

T100

NEW SOFT

FEBBRAIO 1985 L. 10.000

EXTENDED + 32 K
Programmiamo in
Linguaggio Macchina

EXPANDER
Sottoprogrammi in L/M
con l'EXTENDED-BASIC

MERGE per Mini Memory

IN SOFTWARE & SOFTWARE:
EXPANDER - Merge - Rockfeller
Lunar lander - Calorie
Musical Aider...

**Non lasciare solo
il tuo computer**

r. marchetti

 **microcomputer[®]**

 **microcomputer[®]**

la più autorevole rivista del settore

 **microcomputer[®]**

Technimedia
00141 Roma, via Valsolda 135 - tel. (06) 898654 - 899526

SOMMARIO

□ TI BASIC

Gestione files su cassetta, vi sveliamo tutti i segreti

pag. 2-3

□ EXTENDED

È un tema all'ordine del giorno: proteggere il software. Si può anche con l'Extended

pag. 4

□ TI 99/4A

Con il Clock, Bus ed Accessori, continuiamo il viaggio alla scoperta del nostro TI. Nella cassetta la 2ª lezione di «Teach Bit»

pag. 5-6

□ EXTENDED + ESPANSIONE 32 K

Se possedete sia Extended che Exp 32 K eccovi finalmente il modo di programmare in linguaggio macchina

pag. 7-9

□ EXPANDER

Un programma che carica in memoria un set di sotto programmi in linguaggio macchina. Gira in Extended

pag. 9-10

□ PROGRAMMAZIONE

Per la serie «Come costruire un arcade game» è la volta dello «Score Board»

pag. 11-12

□ MINI MEMORY

Parliamo di Standard e Extended utilities

pag. 13-15

□ MERGE

Dopo il MERGE su cassetta per l'Extended basic, è la volta del MERGE per la MINI MEMORY

pag. 16

□ STATISTICA

La correlazione di Rango. Continuate a seguire questa rubrica, presto riuscirete a fare «13».

pag. 17-18

□ IL CASTELLO NOTTE OSCURA

Un classico Adventure sempre vario e coinvolgente. Buona grafica, gira in Basic

pag. 19-20

□ LUNAR LANDER

Un gioco simpatico in Extended

pag. 21

□ ROCKFELLER

Il simpatico corvo dal becco giallo è protagonista di questo gioco in Extended. Ohh...!!

pag. 22-23

□ DIDATTICA

Italiano. Un utile programma per chi vuole usare il TI 99 anche per studiare

pag. 24

□ CALORIE

Calcolate quante calorie «ingurgitate» in un pasto e regolate di conseguenza la vostra dieta

pag. 25

□ MUSICAL AIDER

Provate ad usare il TI 99 come tastiera musicale. È sorprendente!

pag. 26-27

□ PICCOLI ANNUNCI

pag. 28-29

□ LA POSTA

pag. 30

□ COLLABORATE CON NOI

pag. 31-32

□ TI 99 MARKET

pag. 33-34

TI 99 NEWSOFT

ANNO 2 - Numero sei
FEBBRAIO 1985

MENSILE

Direttore responsabile:

ANTONIO C. LOSITO

Coordinatore Editoriale:

VINICIO CIVININI

Direttore Amministrativo:

ENRICO CALLERIO

Art Director:

BEPPE RE-FRASCINI

Illustrazioni:

MARIA TERESA COCOZZA

Segreteria:

EMANUELA CHIAPELLO

CRISTINA FRISON

Redazione:

FIORENZA AURIEMMA

DARIO BAREZZI

Collaboratori:

SERGIO BORSANI,

GIANPAOLO BOTTIN,

DANIELE CATALFAMO,

PAOLO CIVARDI,

STEFANO DELLI PONTI,

FRANCESCO MACALI,

MAURO MATTIAZZI,

EZIO MONTINI,

LEVIO PEROTTI,

RICCARDO ROSSI,

MARCO SQUINTANI,

ALBERTO STRAFILE,

SEBASTIANO TOMASELLO

TI 99 NEWSOFT è una

pubblicazione della **NEWSOFT snc**,

via Stefano Jacini 4 - 20121 MI

Tel. 02/807464 - Telex 324284

REGISTRAZIONE AL TRIBUNALE

DI MILANO N. 323 DEL 7.7.84

© COPYRIGHT NEWSOFT snc

TUTTI I DIRITTI RISERVATI

Manoscritti e foto originali, anche se

non pubblicati, non si restituiscono

ed è vietata la riproduzione anche

parziale di testi, fotografie e listati.

Pubblicità: AV STUDIO

Via Giuseppe Meda, 23

Tel. 02/8357155

Abbonamenti: l'abbonamento annuo

a 11 numeri di **TI 99 NEWSOFT**

costa L. 95.000 per l'Italia; L. 160.000

per l'Europa (spedizione aerea); L.

200.000 per tutti gli altri paesi.

L'importo, per assegno non

trasferibile o per vaglia postale, va

inviato a NEWSOFT snc, via S.

Jacini 4 - 20121 Milano.

Stampa: CGB, Albegno di Treviolo

(BG)

Concessionaria per la distribuzione:

EUROSTAMPA Srl - C.so Vittorio

Emanuele 111 - 10123 Torino - Tel.

011/538166-7

L'editore non si assume

responsabilità alcuna sul contenuto

degli articoli e/o software inviati.

Per gli articoli firmati e/o siglati da

collaboratori esterni, la redazione si

assume la responsabilità prevista

dalle leggi sulla stampa.

Tutti possono collaborare a TI 99

NEWSOFT, sia con articoli che con

programmi. Scriveteci o telefonateci

in redazione per avere tutti i

ragguagli.

downloaded by www.ti99iuc.it

GESTIONE FILES SU CASSETTA

Finalmente, dopo due mesi, siamo arrivati alla parte che più vi interesserà. Questa volta vi proponiamo delle semplici e pratiche routines di facile applicazione, che potranno trovare posto nei vostri programmi, senza creare alcun problema. Se ci avete seguito nelle precedenti puntate della nostra rubrica, non troverete alcuna difficoltà nel capire come sono state realizzate, e sarete in grado di poterle adattare nel migliore dei modi ai vostri programmi.

Le routines che vi proponiamo vi verranno presentate e commentate in ordine crescente di complessità.

Le variabili di matrice di riga 10, possono essere sostituite facilmente. Le abbiamo definite variabili di matrice perché esse hanno la caratteristica di essere in memoria e quindi di poter manipolare (cancelare, sostituire) i dati in esse contenuti.

Nell'inserimento dei dati è sempre consigliabile adottare sistemi di verifica dei dati introdotti, come nell'esempio alle righe 50, 60.

La routine di scrittura non comporta molte difficoltà, ma può essere migliorata ulteriormente, per ridurre il tempo di scrittura.

Sostituendo alle righe 110, 120, 130 le seguenti istruzio-

ni, raddoppieremo il numero di dati che verrà salvato, dimezzando il tempo di scrittura.

```
110 FOR C=1 TO 10 STEP 2
120 PRINT ÷ 1 : A$(C), B(C)
      A$(C+1), B(C+1)
130 NEXT C
```

Se avessimo messo come passo di incremento un numero superiore a 2, avremmo dovuto aggiungere al passo 120 altre variabili, quanti erano i dati da inserire per ogni incremento. Per esempio se il passo di incremento fosse stato 3, il passo 110 avrebbe dovuto avere 3 gruppi di variabili:

```
110 FOR C = TO 10 STEP 3
120 PRINT ÷1:A$(C), B(C),
      A$(C+1), B(C+1), A$(
      C+2), B(C+2)
130 NEXT C.
```

Apertura e salvataggio dei dati contenuti in una matrice

```
10 DIM A$(10),B(10)
20 FOR S=1 TO 10
30 INPUT "NOME COGNOME CLIENTE ":A$(S)
40 INPUT "CODICE CLIENTE ":B(S)
50 INPUT "I DATI SONO ESATTI ?(S-N) ":R$
60 IF R$="N" THEN 30
70 NEXT S
80 REM *****
90 REM ROUTINE DI SCRITTURA
95 REM *****
100 OPEN #1:"CS1",INTERNAL,OUTPUT,FIXED 64
110 FOR C=1 TO 10
120 PRINT #1:A$(C),B(C)
130 NEXT C
140 CLOSE #1
```

Questo continuo aggiungere ha un limite, che è posto da FIXED.

Se il valore del FIXED è posto a 64, noi potremmo aggiungere dati al record sino al raggiungimento di tale lunghezza, in questo caso di 64 caratteri.

Vi ricordiamo che il FIXED può avere un valore da 1 a 192.

Ovviamente, in fase di successiva rilettura dei dati, dovremo tenere conto del modo

in cui sono stati registrati.
Se sono stati scritti più dati consecutivamente in un solo record, dovremo leggerne un ugual numero.

Solitamente si ricopia la routine di scrittura sostituendo la PRINT con una INPUT.

Dovremo anche aprire un altro file con le stesse caratteristiche, ma in modo INPUT. Anche qui, possiamo ricopiare tranquillamente la OPEN sopra indicata, cambiando la scritta OUTPUT con INPUT. Se nel nostro programma sono presenti più FILES con caratteristiche diverse, dovremo aprire tanti file quanti sono i gruppi con caratteristiche uguali.

Per esempio se in un pro-

Esempio di doppia apertura di files input, output

gramma dobbiamo scrivere e leggere dei dati, dovremo aprire due FILES, uno di scrittura e uno di lettura.

Il numero posto dopo il cancelletto «++», serve come riconoscimento, quindi non deve essere uguale a nessun altro FILE già aperto in precedenza.

La grandezza della matrice A\$ può essere aumentata, ma ricordatevi di cambiare anche la lunghezza dei cicli FOR-NEXT delle righe 160, 240, 300

Non ci resta che dire: arriverci alla prossima puntata

EZIO MONTINI

**Avete
acquistato
il Numero
di Febbraio?**

**Ma sapete
che sono usciti
i Numeri di
Giugno
Settembre
Ottobre,
Novembre,
Dicembre e
Gennaio?**

**Per riceverli
direttamente
a casa
vostra,
scriveteci!
Ve li
invieremo in
contrassegno
a L. 10000
(+spese cadauno).**

**TI 99 NEWSOFT
Via S. Jacini, 4
20121 MILANO**

```

70 DIM A$(10)
80 REM *****
90 REM APERTURA DEI FILES
100 REM *****
110 OPEN #1:"CS1",INTERNAL,INPUT,FIXED 64
120 OPEN #2:"CS1",INTERNAL,OUTPUT,FIXED 64
130 REM *****
140 REM INSERIMENTO DATI
150 REM *****
160 FOR S=1 TO 10
170 INPUT "NOME COGNOME CLIENTE ":A$(S)
180 INPUT "I DATI SONO ESATTI ?(S-N)":R$
190 IF R$="N" THEN 140
200 NEXT S
210 REM *****
220 REM SCRITTURA DEI DATI
230 REM *****
240 FOR A=1 TO 10 STEP 2
250 PRINT #2:A$(A),A$(A+1)
260 NEXT A
270 REM *****
280 REM RILETTURA DEI DATI
290 REM *****
300 FOR A=1 TO 10 STEP 2
310 INPUT #1:A$(A),A$(A+1)
320 NEXT A

```

PROTEGGERE IL SOFTWARE

Uno dei tanti problemi sorti negli ultimi anni è il cosiddetto vampirismo di software. Purtroppo, il supporto magnetico costituito dalla cassetta è decisamente vulnerabile, qualunque sia la protezione del programma; basti pensare che ognuno di noi può duplicare fisicamente la cassetta con due registratori collegati in parallelo.

Questo metodo è il più diffuso, ed è impossibile evitarlo. Per nostra fortuna non tutti posseggono una doppia piastra per la duplicazione. Quindi, noi possiamo provare in vari modi ad impedire a quel malcapitato possessore del nostro programma, che ignaro delle tremende protezioni, provi il classico SAVE CS1 o DSK1, che dir si voglia, dopo averlo caricato.

In EXTENDED BASIC, esiste un modo semplice quanto sadico, per impedire non solo il list, ma anche il salvataggio e quindi il cambiamento del programma o la cancellazione di linee alla cieca, es. 1 «enter» ecc...

Per proteggere in tale modo i nostri programmi, basta salvarli in questo modo:

SAVE CS1, PROTECTED

Il programma salvato sulla cassetta, risulta protetto nella maniera sopra descritta

per sempre.

ATTENZIONE: IL METODO DI PROTEZIONE È IRREVERSIBILE.

Assicuratevi di essere in possesso di una copia non protetta, e che il programma sia privo di errori.

Vi raccomandiamo di evitare di spedirci programmi protetti da pubblicare, perché detta protezione ci impedisce di valutare il vostro lavoro, ed ogni programma protetto che riceveremo sarà considerato nullo, quindi da cestinare. Questa protezione tuttavia non impedisce al possessore del programma di mandarlo in esecuzione.

Ci sono altri metodi di protezione meno drastici che possono impedire il list, o il run del programma in un certo modo.

Per esempio noi possiamo avvertire che il programma può essere mandato in esecuzione solo se caricato con una RUN «CS1», in caso contrario si fermerà causando il CRASH o blocco del sistema, costringendo il malcapitato trasgressore a spegnere la macchina con conseguente perdita del programma.

Avrete notato che con il modulo EXTENDED BASIC, ci è consentito leggere il contenuto di alcune locazioni di memoria, diciamo alcune,

perché ci sono delle locazioni che non sono leggibili da BASIC e la lettura accidentale o VOLONTARIA di tali locazioni, porta inevitabilmente al blocco del sistema.

La locazione decimale -26624, se letta da i risultati sopra citati.

La CALL PEEK (-26624, A), se eseguita blocca il sistema. Il modo più semplice per fare capire al computer come è stato mandato in RUN un programma è di controllare l'angolo in basso a sinistra dello schermo.

Se il programma è mandato in esecuzione con una RUN «CS1», troveremo uno spazio vuoto, corrispondente al codice ASCII 32, letto con una CALL GCHAR.

In caso non positivo, passeremo il controllo del programma alla linea dove è situata la malefica CALL PEEK (-26624, A).

Ci dispiace concludere qui il discorso, ma non disperate, nel prossimo numero troverete i modi di protezione del listato e l'applicazione pratica di ciò che avete sino ad ora letto.

EZIO MONTINI



CLOCK, BUS ED "ACCESSORI"

IL CLOCK

Come abbiamo già accennato nel numero scorso, l'unità centrale di elaborazione o CPU deve poter conoscere quando porre fine ad una operazione e cominciare quindi quella seguente. Per ovviare a questo problema, all'interno della macchina è posto un dispositivo temporizzatore: il *clock*. Questo prezioso componente, come suggerisce il nome, compie un'opera di sincronizzazione delle operazioni svolte dalla CPU.

Il *clock* dunque emette impulsi elettrici cadenzati, fornendoli con una commutazione di tensione tra zero e cinque volt effettuata da un cristallo di quarzo. Risulta ora chiaro come si possa, avendo la velocità del segnale, capire la velocità di lavoro della macchina dato che tanto più veloce sarà il susseguirsi degli impulsi tanto più celere sarà la reazione del microprocessore.

I BUS

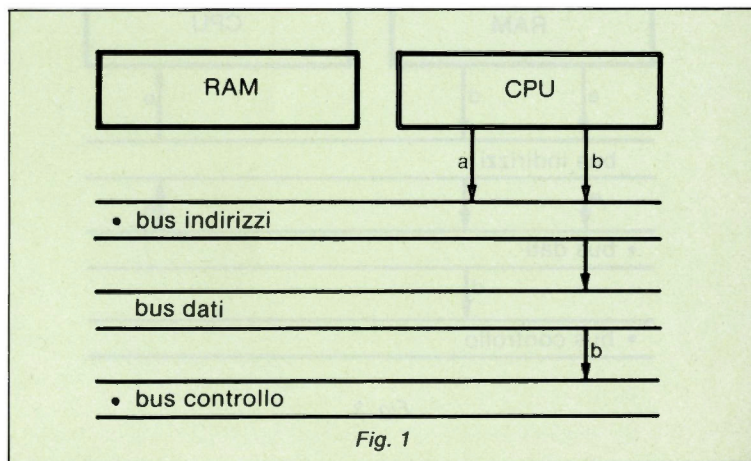
Al pari di una metropoli ben organizzata anche i vari componenti dell'elettronica del nostro TI 99 sono messi in comunicazione tra loro per mezzo di *bus*, che nel nostro

caso sono delle piste conduttrici di impulsi. Tali *bus* sono responsabili del trasferimento dei *bit*-informazione per le varie unità.

In un sistema a microprocessore i *bus* sono generalmente tre: uno relativo ai dati, l'altro per gli indirizzi e un terzo di controllo, che trasferisce le istruzioni dalla memoria alla CPU. I *bus* dati e di controllo codificano l'istruzione trattandola a gruppi di otto *bit*, quello degli indirizzi opera invece a sedici *bit*. Grande importanza hanno poi i *bus* di collegamento delle interfacce, dato che permettono di effettuare collegamenti di periferiche o espansioni senza dover manipolare le connessioni dei circuiti interni del calcolatore.

Interrompere una sola di queste piste comprometterebbe quindi l'intero funzionamento della macchina; poniamo per questo motivo una particolare attenzione sul loro operare, facendo un piccolo esempio di procedura che metta in risalto la sequenza degli eventi.

1) L'unità di controllo della CPU inizia un'operazione di lettura dalla memoria centrale di un dato, posto in una certa locazione. Come prima cosa invia l'indirizzo della locazione alla memoria sfruttando il *bus* indirizzi e, quasi simultaneamente, segnala con il *bus* di controllo che si tratta di un'operazione di lettura della memoria centrale e non magari di scrittura sulla memoria esterna (fig. 1).



2) La memoria centrale comincia a leggere il dato richiesto nella locazione che le è stata fornita segnalando all'unità di controllo di attendere. Per far questo, invia un'istruzione di messa in attesa

lungo il *bus* di controllo, senza la quale la CPU continuerebbe tranquilla a eseguire le operazioni seguenti senza aver ricevuto il dato richiesto. (fig. 2)

3) Non appena la memoria

ACCESSORI

Pur non potendosi definire parte integrante dell'*hardware* di un sistema, alcuni elementi svolgono importantissime funzioni che assicurano il corretto funzionamento della macchina. Uno di questi è, per esempio, l'alimentatore di tensione che, tanto criticato per la sua grossolana estetica a pedale, fornisce al computer la corrente elettrica alla tensione richiesta. Date le sue grosse dimensioni, la Texas ha pensato bene di separarlo dall'unità centrale ponendolo in un contenitore di plastica nero e collegandolo con un primo cavo terminante in una presa a quattro poli (due superiori e due inferiori) da agganciare al computer e un secondo con la spina per la corrente secondo gli standard europei. È inoltre dotato di stabilizzatori per correggere gli eventuali sbalzi di tensione, che danneggerebbero i delicati microcomponenti elettronici del sistema, e di un fusibile da 250 V in caso di improvvise cadute di tensione (vedi black-out).

Poiché l'alimentatore resta sotto tensione anche se il computer è spento, in caso di assenza prolungata sarebbe opportuno staccare la presa di corrente o per i più bravi, inserire un interruttore tra la presa e il nostro "pedalone".

MARCO SQUINTANI

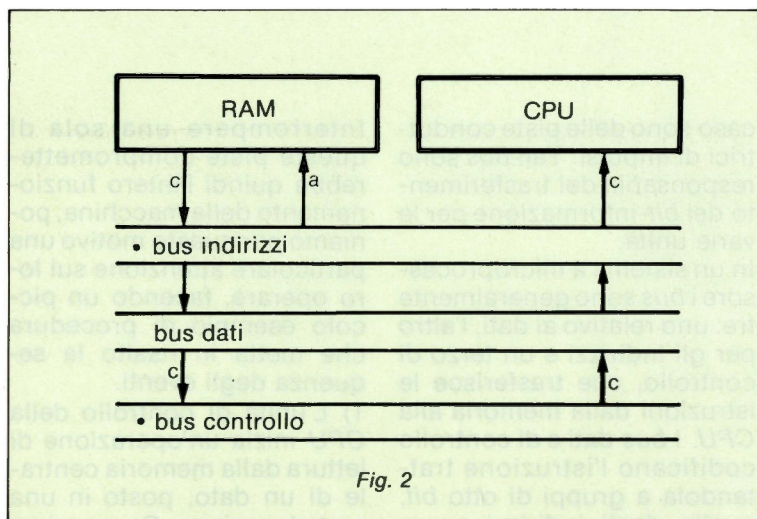


Fig. 2

ha letto il dato nella locazione segnalata, annulla il segnale di attesa e invia l'informazione grazie al *bus* dati.

Finalmente l'unità di controllo ha ottenuto ciò che voleva ed è pronta a procedere all'istruzione successiva. (fig. 3)

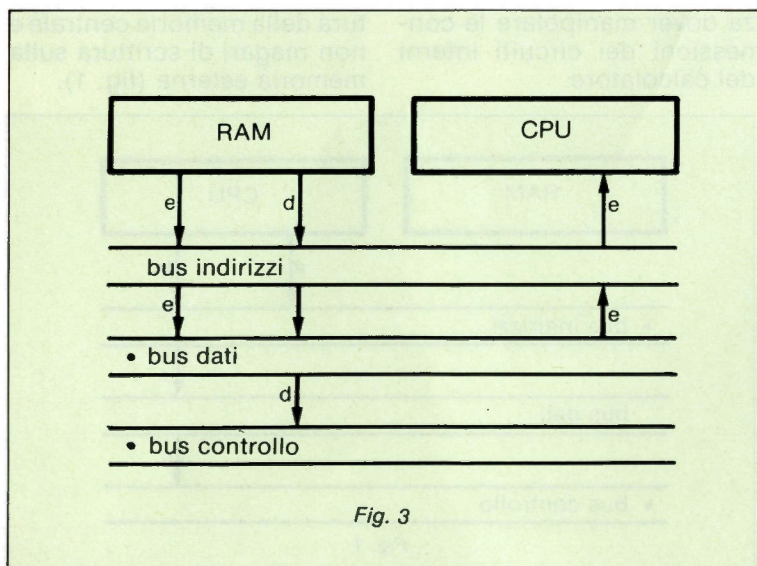


Fig. 3

LINGUAGGIO MACCHINA

Eccoci qui come promesso, e per la gioia degli appassionati della velocità dei programmi, vi insegneremo come programmare in linguaggio macchina con l'ausilio dell'EXT. BASIC + EXP. 32K.

Programmare in linguaggio macchina significa scrivere dei programmi assembly direttamente con delle CALL LOAD.

La programmazione in linguaggio macchina presuppone una minima conoscenza

za dell'assembler del TMS 9900, che come sapete è il nostro processore.

Se ci avete fedelmente seguito nelle rubriche assembly e mini memory, non troverete difficoltà a capire cosa sono le word, i byte, bit, ecc.

Vi consigliamo di andare a rivedere questi concetti base, perché ci risulterebbe un po' difficoltoso ripeterli in questa rubrica.

Il nostro processore è a 16

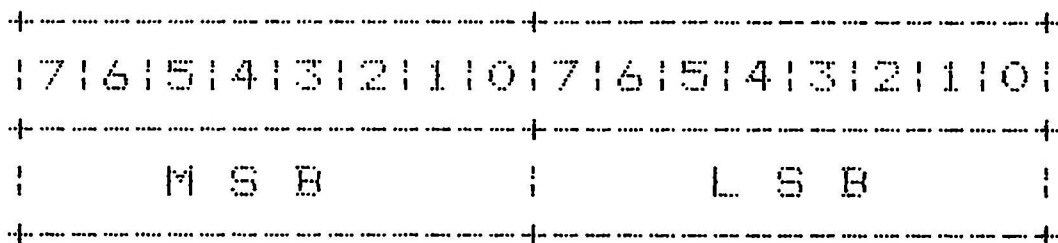
bit, ciò significa che lavora con delle word a 16 bit, a sua volta divisibili in due byte.

Il processore assegna ad ognuno di questi byte un'importanza.

Quello di sinistra è considerato come il MSB (Most Significant Byte), il più importante o significativo.

Quello di destra è considerato come LSB (Less Significant Byte), il meno importante o significativo.

Configurazione di word



Questa distinzione che apparentemente può risultare insignificante, assume molta importanza nel momento in cui noi dobbiamo tradurre un programma da assembly a linguaggio macchina.

Infatti nel MSB troviamo il codice di esecuzione dell'istruzione assembly e nel LSB troviamo il valore da noi inserito per eseguire quella determinata funzione.

Un esempio può essere facilmente dato da:

l'istruzione JMP ha come codice di esecuzione il valore esadecimale >10, detto valore sarà posto nel PSB; il LSB conterrà, espresso in esade-

cimale, il numero di locazioni da saltare, dato che JMP in assembly rappresenta un salto incondizionato di n. locazioni, si può paragonare al GOTO presente nel linguaggio BASIC.

Dato che il programma in LM deve essere inserito in memoria con delle CALL LOAD, e dato che le CALL LOAD non accettano valori esadecimali, dovremo prenderci la briga di tradurre pazientemente, istruzione per istruzione, tutto il programma assembly che vogliamo trasformare, tenendo conto che il MSB e il LSB, vanno tradotti separatamente in decimale.

Per esempio la word esadecimale >1003, dovrà essere tradotta >10 e >03, con conseguente traduzione in decimale 16,3. I valori così ottenuti dovranno essere inseriti in memoria consecutivamente ed in progressione di locazione.

In EXTENDED BASIC, possiamo cominciare ad introdurre i nostri programmi in LM dalla locazione decimale 12288, per evitare problemi. È tuttavia possibile cominciare ad introdurre dati da locazioni posteriori, cioè minori, a quella indicata, ma ciò richiede accurati calcoli per determinare la qualità di me-

moria che il computer richiede per il buon funzionamento del nostro programma in Linguaggio Macchina.

COME OTTENERE LA TRADUZIONE DA ISTRUZIONE ASS. A ESADECIMALE

Ci sono tre metodi per ottenere questa traduzione:

- 1) Possedere il modulo MINI MEMORY e ricopiare la traduzione scritta all'immediata sinistra dell'istruzione.
- 2) Essere in possesso del modulo ED. ASS. e fare un assemblaggio con codice oggetto e sorgente.
- 3) Conoscere l'assembly e tradurre a tavolino le istruzioni.

Il terzo modello presuppone che si sia in possesso del faticoso manuale dell'ED. ASSEMBLER, dove sono contenute tutte le traduzioni delle istruzioni.

Noi cercheremo di aiutarvi pubblicando delle tabelle di traduzione, istruzione per istruzione.

Vi ricordiamo che i programmi scritti per l'ED. ASS. o per MINI MEMORY, non sempre possono girare in ambiente EX. BASIC. Le differenze principali sono date dalla diversa locazione delle utility richiamabili con BLWP.

Alleghiamo una tabella di traduzione di utility da MINI MEMORY a EX. BASIC.

Una volta inserito il programma in LM, dovremo avvertire il computer che il nostro programma è pronto per essere eseguito.

Per poter fare ciò è necessario compiere due operazioni. Con due CALL LOAD andremo ad inserire il nome del

+-----+ !TABELLA DI CONVERSIONE DA ED.ASS! !O MINI MEMORY A EXT.BASIC + 32K! +-----+		
BLWP nome utility	valore BLWP ed.ass,m.mem	valore BLWP ext.bas.+32K!
XMLLNK	>601C	>201B
KSCAN	>6020	>201C
VSBW	>6024	>2020
VMBW	>602B	>2024
VSBR	>602C	>202B
VMBR	>6030	>202C
VWTR	>6034	>2030
NUMASG	>6040	>200B
NUMREF	>6044	>200C
STRASG	>604B	>2010
STRREF	>604C	>2014
ERR	>6050	>2034

programma nella ref/def table e diremo al computer dove si trova l'inizio e la fine del programma.

Per prima cosa dovremo agire sulla locazione decimale 8194, introducendo due numeri che rappresentano la locazione di fine del programma.

Poi di seguito introdurremo due valori fissi che sono 63,

240.

I numeri di fine programma si ottengono dividendo la locazione finale per 256, poi moltiplicando il decimale per 256.

Se il risultato non ha decimale ed è quindi un numero intero, introdurremo come secondo numero uno zero.

Esempio: end loc. 12561, i numeri saranno:

12561/256=49.07, il primo numero è 49, poi 0.07*256=18, il secondo è 18.

La CALL LOAD sarà: CALL LOAD (8194, 49, 18, 63, 240) Poi dovremo dare un nome al programma per farlo eseguire, chiamandolo con una CALL LINK.

Questa volta dovremo agire sulla locazione decimale 16368, qui dovremo inserire consecutivamente dapprima il nome del programma sotto forma di codice ascii, poi due numeri che rappresentano l'indirizzo della prima istruzione eseguibile in LM. Detti numeri si ottengono operando sulla locazione di memoria della prima istruzione eseguibile, con il metodo sopra indicato, seguendo le stesse regole.

Il nome del programma non deve superare i sei codici ascii 32 sino al raggiungimento di sei numeri, e questa regola va sempre osservata rigorosamente.

Esempio: il nome che vogliamo assegnare al programma è PROVA e la locazione dove si trova la prima istruzione eseguibile è 12288. La CALL LOAD che dovremo avere sarà: CALL LOAD (16368, 80, 82, 79, 86, 65, 32, 48, 0).

Per questa volta abbiamo concluso, nella cassetta troverete un programma di nome EXPANDER che tramite il linguaggio macchina fornisce sei nuove istruzioni al BASIC. Oltre che servirvi come esempio, vi sarà molto utile nella programmazione di giochi.

EZIO MONTINI



«EXPANDER»

Il programma di nome EXPANDER che trovate sulla cassetta, deve essere caricato con la formula OLD CS1, poi mandato in esecuzione con il classico RUN.

Questo programma carica in memoria un set di sottoprogrammi in linguaggio macchina, eseguibili con delle CALL LINK.

Una volta caricati in memoria, dopo che il RUN è stato eseguito, i sottoprogrammi rimangono memorizzati, anche dopo aver dato il NEW.

Gli unici modi per cancellarli è di eseguire una CALL INIT, un BYE, un QUIT e ovviamente di spegnere il computer.

I sottoprogrammi aggiunti sono sei, rispettivamente divisi in due gruppi. Il primo gruppo è costituito da tre sottoprogrammi, che hanno la funzione di memorizzare.

1) CALL LINK («SAVES»)

Questa istruzione permette di salvare un'intera pagina grafica, permettendone la successiva utilizzazione.

Come potrete constatare dalla sorgente del programma assembler, riportata in figura, il sottoprogramma in questione salva la mappa video, locata dalla locazione decimale 0 alla locazione decimale 768. Introducendone il contenuto in un apposito buffer chiamato SCR.

2) CALL LINK («SAVEP»)

Detto sottoprogramma salva la mappa dei pattern o caratteri, locata dalla locazione esadecimale >03FO e prosegue per 912 BYTES, corrispondente a 8 bytes per carattere; i caratteri sono 114, equivalente a 30-143 (30 compreso).

Sono memorizzati in un buffer di 912 bytes, chiamato PAT.

3) CALL LINK («SAVEC»)

SAVEC permette di salvare l'intera mappa dei colori, locata a partire dalla locazione esadecimale >0800 e che prosegue per 32 bytes, equivalenti a 2 bytes per ogni gruppo di caratteri.

Sono memorizzati in un buffer di 32 bytes chiamato COL. Il secondo gruppo è anch'esso costituito da tre sottoprogrammi aventi la specifica funzione di richiamare in memoria i dati salvati precedentemente negli appositi buffers.

1) CALL LINK («LOADS»)

Richiama dal buffer di nome SCR i dati della pagina grafica.

2) CALL LINK («LOADP»)

Richiama dal buffer di nome PAT i dati della mappa dei caratteri.

```
*****
*
*   ESTENSIONE ISTRUZIONI EX.BASIC+32K
*
*   by EZIO MONTINI and MARCO CONTENTE
*
*       version 1.0
*
*   I sottoprogrammi sono richiamabili
*   direttamente da EX.BASIC+32K con
*   CALL LINK .
*   Le CALL LINK possibili sono sei :
*
* CALL LINK("SAVES") CALL LINK("LOADS")
* CALL LINK("SAVEP") CALL LINK("LOADP")
* CALL LINK("SAVEC") CALL LINK("LOADC")
*
*****
```

```
DEF SAVES,SAVEP,SAVEC
DEF LOADS,LOADP,LOADC

SCR  BSS  768
PAT  BSS  912
COL  BSS  32

EVEN

SAVES  MOV  R11,R10
      LI   R0,0
      LI   R1,SCR
      LI   R2,>02FF
      BLWP @>202C
      MOV  R10,R11
      B    *R11
SAVEP  MOV  R11,R10
      LI   R0,>03F0
      LI   R1,PAT
      LI   R2,912
      BLWP @>202C
      MOV  R10,R11
      B    *R11
SAVEC  MOV  R11,R10
      LI   R0,>0800
      LI   R1,COL
```

```
LI   R2,32
BLWP @>202C
MOV  R10,R11
B    *R11
LOADS MOV  R11,R10
      LI   R0,0
      LI   R1,SCR
      LI   R2,>02FF
      BLWP @>2024
      MOV  R10,R11
      B    *R11
LOADP MOV  R11,R10
      LI   R0,>03F0
      LI   R1,PAT
      LI   R2,912
      BLWP @>2024
      MOV  R10,R11
      B    *R11
LOADC MOV  R11,R10
      LI   R0,>0800
      LI   R1,COL
      LI   R2,32
      BLWP @>2024
      MOV  R10,R11
      B    *R11
      END
```

3) CALL LINK («LOADC»)

Richiama dal buffer di nome COL i dati di mappa dei colori.

Vi accorgerete come questo semplice programmino in LM migliorerà i vostri giochi, permettendo immediati cambi di schermo, pattern, colori, con il solo comando CALL LINK, specificando il nome della utility da voi desiderata.

Le CALL LINK possono essere eseguite sia in modo comando che in modo esecutivo. Il programma in EXT. BASIC, contenuto nella cassetta, potrà facilmente trovare posto nei vostri programmi date le sue ridotte dimensioni e il suo scarso utilizzo di memoria (Kb utilizzati=1.230). Alleghiamo un piccolo programmino di esempio per mostrarvi l'enorme potenzialità del nostro programma. Prima di eseguire il programma di esempio, assicurarsi che il programma EXPANDER sia già stato caricato in memoria.

EZIO MONTINI

COME COSTRUIRE UN ARCADE GAME: GLI SCORE BOARD

Ogni volta che giochiamo con un videogame vorremmo riuscire a battere il record del gioco e non a caso si fanno gare, si cerca di battere record di durata e resistenza, insomma ci piace la competizione, tutto questo è reso possibile, dal gioco ed ovviamente da il contapunti, per intenderci lo «score».

Se qualcuno ricorda, i primi giochi si limitavano a segnare i punti fatti nella partita, solo in seguito cominciavano a tenere memorizzato anche il punteggio più alto realizzato e poi il massimo lo «Score board» cioè il tabellone che richiamava i migliori 10 punteggi con il nome o le iniziali di chi lo realizzava.

Dopo questa breve storia, passiamo alla nostra solita routine. Infatti questa puntata è dedicata alla creazione di uno «Score board», ed alla visualizzazione dei punteggi. Anche per i punti è necessario preparare una routine a parte mentre il richiamo della sudetta dovrà avvenire solitamente dalla routine di Distruzione Nemico, che vedremo in una delle prossime puntate, dove viene dato il consenso ad incrementare il settore punti.

Ovviamente in TI BASIC, non

disponendo dell'istruzione «DISPLAY AT», è necessario creare anche una routine di aggiornamento punti che permetta la stampa del punteggio al suo posto senza effettuare lo scorrimento dello sfondo.

Quando preparerete il quadrato di gioco sarà opportuno creare una stringa che servirà per stampare i punti, che potrebbe essere impostata in questo modo:

```
370 HPUNTI$="0000"
380 PUNTI$="0000"
390 P$="SCORE:" & PUNTI$
400 R=1
410 C=3
420 GOSUB 5000
430 P$="HI-SCORE:"
    &H PUNTI$
440 R=1
450 C=18
460 GOSUB 5000
```

Queste poche righe servono per creare la scritta azzerata del contatore punti in cima al quadro di gioco, i vettori «R» e «C», sono le coordinate di partenza delle scritte mentre alla riga 5000, dove ci manda la «GOSUB», c'è la routine che chiameremo di «PRINT AT».

Ed ora il listato «PRINT AT»:

```
5000 REM PRINT AT
5010 Z=LEN(P$)
5020 FOR CF = 1 TO Z
5030 CALL HCHAR(R,C+CF,
    ASC(SEG$(P$,CF, 1)))
5040 NEXT CF
5050 RETURN
```

Ovviamente questa routine può servirvi anche ad altri scopi. Basta dare un valore alla «RIGA» ed alla «COLONNA», definire la Stringa, ed essa verrà stampata a partire dalle posizioni da voi definite.

Una volta analizzato il gioco, ovviamente bisognerà creare una tabella punti affinché il computer assegni ad ogni nemico abbattuto il punteggio corrispondente.

I metodi sono molteplici, noi ve ne elenchiamo uno dei tanti:

Creazione matrice
contenente i punteggi

```
10 DIM A (4)
20 FOR C=1 TO 4
30 A(C)=C*100
40 NEXT C
```

```
3000REM NEMICO COLPITO
```

GIOCHIAMO CON IL TI 99/4A



Un libro che non deve assolutamente mancare nella vostra libreria.

Potrete riceverlo a casa vostra a sole **L. 9500** + Spese di spedizione.

Per ogni richiesta di contrassegno, utilizzate il tagliando qui sotto spedendolo a:

NEWSOFT snc
via S. Jacini, 4
20121 MILANO

Vogliate spedirmi in contrassegno di L. 9500 + spese il libro:
«GIOCHIAMO CON IL TI 99/4A»

Cognome

Nome

Indirizzo

downloaded by www.ti99iuc.it

Cap. Loc.

```
3090 AS=X — Valore ASCII carattere nemico colpito
3100 AS=AS-127 — Sottrarre il numero ASCII dell'ultimo
3105 REM ROUTINE PUNTI — carattere prima di quelli nemici
3110 PUNTI=PUNTI+A(AS)
3120 P$=STR$(PUNTI) — Determinazione COLONNA DI PARTENZA STAMPA
3140 GOSUB 5000
3150 P$=" "
3160 C=O
3170 RETURN
```

Questa routine quindi permette di stampare i punti al posto prestabilito, ovviamente i nemici, o meglio, i caratteri definiti, dovranno essere consecutivi e cominciare dal carattere 128 in modo che (in questo esempio sono stati considerati 4 nemici diversi), quando viene effettuata la sottrazione alla linea 3100 i numeri possibili sono dall'1 al 4, coprendo così l'intera matrice contenente i punti. Se i punteggi invece non sono differenziati, ma uguali per tutti i nemici si possono evitare tanti intoppi eliminando le righe da 10 a 40 e le righe da 3090 a 3100 ed inserendo al posto della riga 3110 la seguente riga:

```
3110 PUNTI=PUNTI+100
```

Qui direi di sospendere la chiacchierata sul calcolo dei punti, anche perché il discor-

so molto generico si presta a tante varianti, tuttavia ritengo che la routine che vi abbiamo presentato possa essere interessante e anche piuttosto veloce.

Ora invece per i perfezionisti vedremo di presentare una routine di memorizzazione dei record e delle iniziali, fedeli alle tradizioni delle più famose Software-House (la trovate nella cassetta).

Anche qui il discorso si presta a tante variazioni sul tema e perciò abbiamo pensato di rivolgere la nostra attenzione al tipo più usato, cioè a quello che elenca 5 punteggi e quindi con il movimento del Joystick (nel nostro caso anche da tastiera) o a destra o a sinistra, c'è lo scorrimento in un riquadro del Set Alfabetico, e quando la lettera interessata appare, basta premere Enter o il tasto "FIRE" e la lettera appare al proprio posto.

MMG

MAURO MATTIAZZI

STANDARD E EXTENDED UTILITIES

Nel numero di gennaio abbiamo cominciato a parlare della ROOM del nostro modulo: abbiamo preso in considerazione i vettori e la NAME LINK ROUTINE. Si è fatto, in questo modo, qualche passo in avanti nella comprensione del funzionamento «interno» del computer (a livello software). Questo mese ci occuperemo in dettaglio delle più importanti tra le extended e standard utilities, necessarie per la programmazione in Assembler.

GPLLNK: consente al programmatore di utilizzare le routines in GPL, residenti nella GROM. Possiamo così inizializzare il sistema, caricare i diversi sets di caratteri, eseguire operazioni matematiche, trigonometriche, etc. Per fare ciò è sufficiente dare un «BLWP >6018» (che è l'indirizzo del vettore della GPLLNK), seguito da un DATA con un numero che identifica la GPL routine da far eseguire.

Un esempio:

```
BLWP >6018
DATA >0036
```

Il computer esegue un suono di warning (il solito). All'indirizzo >837C troviamo

il GPL STATUS BYTE: esso fa da STATUS REGISTER mentre sono in esecuzione le routines in GPL. Il bit 2 (condition bit) è settato dalla DSR routine per indicare che un file non esiste.

Ricodiamo che è opportuno azzerare lo STATUS BYTE prima di chiamare una delle GPL routines per evitare che ci venga segnalato un errore privo di significato.

Nella locazione >8354 è indicato il tipo di errore che eventualmente capita durante l'esecuzione delle floating-point routines, secondo un codice descritto a pagina 42 del manuale MM.

Fra le varie routines richiamabili, segnaliamo la CNS, identificata dal codice >14: essa ci permette di scrivere numeri sullo schermo poiché li converte in stringhe ASCTI. Il numero da convertire deve essere memorizzato in FAC nel formato floating-point. Spendiamo due parole riguardo a questo formato: i numeri floating-point (in italiano: «in virgola mobile»), che noi usiamo abitualmente programmando in linguaggi evoluti come il BASIC, non sono direttamente «masticabili» dalla CPU, che lavora

con gli integers (numeri interi da 8 o 16 bits — > FF o >FFFF).

È questo il motivo per cui troviamo nella ROM delle routines che ci permettono operazioni con i numeri in virgola mobile. Il numero è contenuto in 8 bytes, secondo la notazione «radix 100» ($A \cdot 100^{\Delta B}$): il primo byte indica l'esponente, incrementato di >40, a cui elevare 100 (è quindi $B + >40$). Ciò significa che se ha un valore di >43 l'esponente è 3; con un valore di >3A è -5.

Per indicare che un numero è negativo i primi due bytes sono espressi nella forma two's-complement.

Gli altri sette bytes contengono A convertito nella notazione esadecimale; in particolare il secondo byte rappresenta il valore della parte del numero espresso in «radix 100 notation».

XMLLNK: essa ci permette il linking con le ROM-routines. Ci sono due modi per indicare quale routine si vuole far eseguire: nella word (che chiameremo word A) di seguito all'istruzione «BLWP >601C» si può memorizzare, tramite lo statement DATA, il codice d'identificazio-

ne (ID code) oppure l'indirizzo di partenza della routine. Parliamo del codice: con esso possiamo riferirci nominalmente ad una delle 16 tabelle presenti in memoria, ognuna delle quali contiene 16 indirizzi che rappresentano gli entry points delle routines. Il codice viene memorizzato nel byte più significativo della word A; il nybble alto indica la tabella a cui ci si vuole riferire. Il secondo nybble indica la word che interessa nella tabella specificata.

Solo 8 (da >0 a >7) sono le tabelle utilizzabili, poiché quando è settato il bit 0 della word A, allora il contenuto è considerato un indirizzo: è questo il secondo metodo di linking. Nella figura 1 sono indicati gli indirizzi delle tabelle.

Facciamo notare che la CIF (con codice >23) è inutilizzabile: il manuale Editor-Assembler ci avverte che essa è caricata nell'espansione di memoria dell'E-A Loader (la tabella 2 è all'indirizzo >2000).

Tuttavia è stata inserita nella ROM del modulo all'indirizzo >618C; per utilizzarla è sufficiente digitare queste istruzioni:

```
BLWP @>601C
DATA >7200
```

KSCAN: questa routine chiama a sua volta la SCAN nella consolle, situata all'indirizzo >000E.

Un accesso diretto alla SCAN è possibile, ma bisogna ricordarsi di caricare (load) il GPL Workspace (operazione svolta dalla KSCAN). Questa effettua una scansione della tastiera (come il

Numero tabella	Indirizzo
> 0	Specificato in > OCFA
> 1	Specificato in > OCFC
> 2	> 2000
> 3	> 3FC0
> 4	> 3FE0
> 5	> 4010
> 6	> 4030
> 7	> 6010

Figura 1 - Gli indirizzi delle tabelle.

```

10 CALL KEY (O, R, S)      CLR RO
20 IF R<83 THEN 10        MOV RO, @>8374
30 END                     A BLWP @>6020
                           MOV @>8375,1
                           CI R1, >5300
                           JNE A
                           MOV RO, @>837C
                           B *R11
                           B *R11

```

Unità tastiera: zero
Scansione
Lettura Codice ASCII

Pulisce lo Status
Byte
Ritorna

Figura 2 - Esempio KSCAN.

CALL KEY del BASIC), rilevando il tasto premuto e aggiornando lo STATUS (sempre >837C).

Nella locazione >8374 viene specificata l'unità di tastiera; in >8375 viene memorizzato il codice ASCII del tasto premuto (>FF se nessun tasto è stato premuto); in >8376->8377 è indicata la posizione dei joystick nel solito modo (>04, >00, >FC). Il bit 2 (e non 5) dello STATUS è settato quando è stato premuto un tasto diverso dal tasto premuto durante l'ultima scansione con la routine. Nella fig. 2 vedete un esempio di utilizzazione della routine, confrontato con un frammento di programma in BASIC. Bisogna ricordarsi di azzerare sempre lo STATUS BYTE: se, ad esempio, tralasciate l'istruzione «MOVB RO, @>837C»el programmi-

no, il computer vi segnalerà «ERROR CODE 00».

VSBW, VMBW, VSBR e VMBR: sono quattro utilissime routines che ci facilitano il compito di scrivere e leggere nelle locazioni della VDP RAM (e quindi sullo schermo). Nella fig. 3 trovate i disassemblati di queste, con relative spiegazioni.

Come potete vedere, esse utilizzano un'unica subroutine (>616C) che scrive l'indirizzo di scrittura o di lettura nell'apposita locazione (>8C02-VDPWA). Ricordiamo che in essa non è possibile passare un word-address: è necessario muovere dapprima il byte meno significativo e successivamente (deve passare qualche microsecondo) il più significativo.

Le routines di lettura utilizzano la subroutine a partire da >6174: le prime due istruzio-

■ VSBW		
> 6126	BL &616C MOV &>0002 (R13), &>8C00 RTWP	Scrive un singolo byte (>8C00 è il VDPWD) R13 dopo un BLWP contiene il WP. &>0002 (R13) rappresenta il vecchio R1.
■ VMBW		
> 6132	BL &>616C	Ciclo di scrittura. Il VDP RAM WRITE ADDRESS è autoincrementante
> 6136	MOV *R1 +, &>8C00	
	DEC R2 JNE >6136 RTWP	
■ VSBR		
> 6040	BL &>6172 MOVB &>8800, 0002(R13) RTWP	Legge e muove in R1 (>8800 è il Registro di Lettura).
■ VMBR		
> 614C	BL &>6172	Ciclo di Lettura. Anche il Registro di Lettura è autoincrementante
> 6150	MOVB &>8800, *R1+	
	DEC R2 JNE >6150 RTWP	
■ SUBROUTINE		
> 616C	LI R1, >4000	Per il set bit 1 con SOC 1, 2 routine di scrittura.
	JMP >6174 CLR R1	Vecchio RO in R2 Scrive il byte meno significativo di R2 (vecchio RO) nel registro di scrittura indirizzo.
> 6174	MOV *R13, R2 MOVB &>7097, &>8C02	
	SOC R1, R2 MOVB R2, &>8C02 MOV &>0002 (R13), R1	Per le routine con multiple bytes.
	MOV &>0004 (R13), R2 B *R11	Ritorno.
■ VWTR		
> 615A	MOV *R13, 1 MOVB &>0001 (R13), &>8C02 ORI R1, >8000 MOVB R1, &>8C02 RTWP	Vecchio RO in R1. Passaggio valore. Set registro (bit 0). Passaggio Registro.

ni sono destinate alle routines di scrittura. Infatti il bit 1 del most-significant byte dell'indirizzo deve essere settato per indicare un'operazione di scrittura e ciò è eseguito dalle istruzioni.

LI R1,>4000
SOC R1,R2

Ripetiamo ora qualcosa che è indispensabile conoscere per riuscire a scrivere sullo schermo con un programma in LM chiamato da BASIC. Per fare questo NON bisogna utilizzare i soliti codici ASCII, ma si deve aggiungere >60 o 90 ad essi (lo si scopre a pag. 61 del manuale MM dove si legge: «...from BASIC there is a SCREEN BIAS of >60. SCREEN VAL=ASCII VAL+>60). Nel manuale E-A si parla di «SCREEN OFFSET of >60».

L'ultima utilities di cui ci occuperemo in questo numero è la VWTR. Ci serve per cambiare i valori degli otto VDP WRITE-ONLY REGISTERS. Dal disassemblato si può vedere che non appena è stato «spedito» il byte meno significativo (che contiene il valore al quale si vuole portare il registro specificato dal primo byte), il bit 0 di quest'ultimo è settato per indicare che non viene spostato un indirizzo. Con questo vi salutiamo: il prossimo mese continueremo la rassegna.

STEFANO
DELLI PONTI

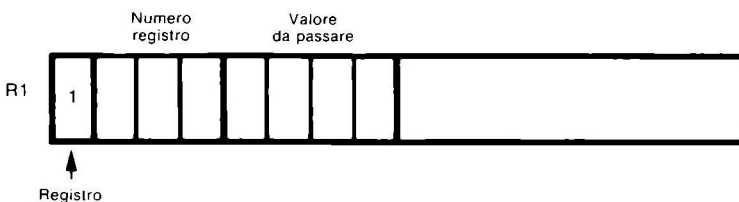


Figura 3

ABBONATEVI
A TI 99
NEWSOFT.
CON SOLE
L. 95.000
RICEVERETE
11 NUMERI
DELLA RIVISTA
E IN PIU'
AVRETE
DIRITTO AGLI
SCONTI SUL
SOFTWARE
E
L'ARDWARE
OFFERTO DA
TI 99 MARKET

Per abbonarsi
basta inviare
un assegno
non trasferibile
o un vaglia
postale a:

NEWSOFT SNC
Via S. Jacini, 4
20121 MILANO

MERGE PER MINI MEMORY

Nel numero di Dicembre abbiamo pubblicato il merge per extended basic ripromettendoci di pubblicare la versione per mini memory in questo numero.

Questo merge al contrario di quello per extended, fonde direttamente il programma in memoria con un programma memorizzato su nastro. Il merge (in effetti il termine "merge" implica anche un operazione di sort numeri di linea, cosa che questo programma non effettua) accoda il programma in memoria ad un programma su nastro nel momento in cui questo viene caricato in memoria.

Questo significa che prima del merge avremo sulla cassetta la prima parte del programma, mentre in memoria avremo la seconda parte. Per effettuare il merge dovremo battere:

CALL LINK («MERGE»)
eseguire le istruzioni sul video.

Facciamo un esempio:

Abbiamo due programmi
10 PRINT «PRIMO PROGRAMMA»

e

20 PRINT «SECONDO PROGRAMMA»

La sequenza per unirli è la seguente:

1) caricate, se non lo avete già fatto, il merge con il comando «L» dell'easy bug

2) torniamo in basic e carichiamo il secondo programma

20 PRINT «SECONDO PROGRAMMA»

3) battiamo CALL LINK («MERGE»), sul video compariranno le istruzioni per il registratore. Carichiamo a questo punto il programma che si trova sul nastro, quando il cursore ricomparirà sul video il merge tra due programmi sarà stato effettuato. Se non ci credete provate a listare il programma.

Contrariamente all'altro programma non esistono particolari limitazioni riguardo la lunghezza dei singoli programmi, comunque bisogna tener presente che la somma della lunghezza dei due programmi, più la lunghezza della tavola dei numeri di linea del secondo programma non deve superare i 14 Kb di memoria, in caso contrario il comando viene abortito.

Con questo abbiamo momentaneamente colmato uno dei buchi più grossi nel basic del TI. Se avete altri problemi non esitate a scriverci, nel limite del possibile vedremo di realizzare le vostre richieste.

RICCARDO ROSSI

CORRELAZIONE DI RANGO

Nella puntata precedente abbiamo pubblicato un programma che ci permetteva, avendo a disposizione dati numerici, di trovare la correlazione esistente tra due serie di dati X e Y.

Qualora non si disponga di precisi dati campionari, per esempio nel caso di valutazioni svolte da giudici diversi chiamati a dare valutazioni soggettive su una prestazione sportiva, su un concorso o, semplicemente, nel caso debbano decidere quale bevanda abbia un gusto migliore, quale detersivo sia più efficace, ecc. sarà necessario ricorrere al calcolo del coefficiente di Spearman.

Nota per i sistemisti al totocalcio:

Avrete notato che il lunedì, dopo la partita, i giudizi sui singoli giocatori delle squadre variano enormemente da giornale a giornale. Applicando il programma che stiamo per mostrarvi avremo modo di assegnare pesi diversi ai giudizi dei giornalisti, farci un'idea più chiara circa lo stato di forma di ciascuna squadra e, magari, scartare valutazioni perennemente difettose o troppo iperboliche.

Il coefficiente di Spearman

Per iniziare i lettori alla comprensione del programma possiamo cominciare con un esempio.

Due giornalisti sportivi hanno valutato le 16 squadre di serie A.

Sommando i valori dei voti dati ai giocatori di ciascuna squadra scopriamo che le squadre, chiamiamole A, B, C, D, E, F, G, H, I, L, M, N, O, P, Q, R per semplicità, sono state classificate in modo diverso dai due giornali.

Forniamo una tabella semplificativa

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1° giornale	B	F	C	A	E	D	G	N	L	I	R	P	Q	O	M	H
2° giornale	C	B	F	G	E	A	D	Q	R	M	L	P	N	I	O	H

Considerazioni

A prescindere dal fatto che le valutazioni sono manifestamente diverse, e questo già lo pensavamo, il pallone è rotondo, e variabile è anche il modo di interpretare il calcio, appare chiaro che, nel caso in cui nessuna squadra fosse stata classificata nella stessa posizione, allora i giornalisti non si sarebbero trovati d'accordo neanche a Natale, come a volte succede, e per noi il discorso statistico si sarebbe concluso.

Abbiamo invece, in tre casi, gli stessi piazzamenti, ciò non significa che di fronte a

giudizi così unanimi la squadra H, per esempio, sarà quella che, concordemente condannata, finirà dritta in serie B, ma che, almeno, è quella che manifesta, per il momento, il grado di forma peggiore, e di ciò potremmo tenere qualche conto statistico.

Ho detto potremmo in quanto prima è, come al solito, indispensabile verificare che le coincidenze non siano avvenute per caso, poiché, allora, ritorneremo al discorso precedente del Natale: lasciamo perdere, che è meglio!

Se non voltate pagina, ora controlliamo.

(Per i non portati alla matematica: saltate pure qualche riga ora, tanto poi si capisce lo stesso).

Il coefficiente di Spearman rho è calcolato dalla formula:

$$\rho = 1 - \frac{6 \sum D^2}{n(n^2 - 1)}$$

dove

D è la differenza tra ranghi per ogni elemento

$\sum$ è la somma dei quadrati di D

n è il numero dei soggetti

rho può variare, come al solito, da +1 a -1; intendendo con +1 una relazione diretta, con 0 la mancanza di qualsiasi correlazione, e con -1 una relazione inversa. La tabella in alto illustrerà meglio il metodo di calcolo.

certa uniformità di giudizio, e questo era quanto ci necessitava stabilire.

PAOLO CIVARDI

squadra	giornale 1	giornale 2	differenza di rango D	D ²
A	4	6	-2	4
B	1	2	-1	1
C	3	1	2	4
D	6	7	-1	1
E	5	5	0	0
F	2	3	-1	1
G	7	4	3	9
H	16	16	0	0
I	10	14	-4	16
L	9	11	-2	4
M	15	10	5	25
N	8	13	-5	25
O	14	15	-1	1
P	12	12	0	0
Q	13	8	5	25
R	11	9	2	4
Totali			0	120

Applichiamo la formula risolutiva di rho:

$$\rho = 1 - \frac{6 \times 120}{16(16^2 - 1)} = 0.8236$$

Questo valore 0.8236 ci è necessario per stabilire la possibilità che una siffatta correlazione sia avvenuta per caso.

Dall'osservazione della tabella allegata notiamo che per 16 campioni, la possibilità al 2.5% è uguale a 0.5000. Poiché 0.8236 è maggiore di 0.5000 possiamo affermare che esistono meno del 2.5% di possibilità che una correlazione così buona, o migliore, sia avvenuta per caso. Possiamo pertanto ritenere che i voti siano stati assegnati con competenza, e con una

Probabilità che rho avvenga per caso

	20%	10%	5%	2%	1%	0.2%
4	.8000	.8000				
5	.7000	.8000	.9000			
6	.6000	.7714	.8286	.8857	.9429	
7	.5357	.6786	.7450	.8571	.8929	.9643
8	.5000	.6190	.7143	.8095	.8571	.9286
9	.4667	.5833	.6833	.7667	.8167	.9000
10	.4424	.5515	.6364	.7333	.7818	.8667
11	.4182	.5273	.6091	.7000	.7455	.8364
12	.3986	.4965	.5804	.6713	.7273	.8182
13	.3791	.4780	.5549	.6429	.6978	.7912
14	.3626	.4593	.5341	.6220	.6747	.7670
15	.3500	.4429	.5179	.6000	.6536	.7464
16	.3382	.4265	.5000	.5824	.6324	.7265
17	.3260	.4118	.4853	.5637	.6152	.7083
18	.3148	.3994	.4716	.5480	.5975	.6904
19	.3070	.3895	.4579	.5333	.5825	.6737
20	.2977	.3789	.4451	.5203	.5684	.6586
21	.2909	.3688	.4351	.5078	.5545	.6455
22	.2829	.3597	.4241	.4963	.5426	.6318
23	.2767	.3518	.4150	.4852	.5306	.6186
24	.2704	.3435	.4061	.4748	.5200	.6070
25	.2646	.3362	.3977	.4654	.5100	.5962
26	.2588	.3299	.3894	.4564	.5002	.5856
27	.2540	.3236	.3822	.4481	.4915	.5757
28	.2490	.3175	.3749	.4401	.4828	.5660
29	.2443	.3113	.3685	.4320	.4744	.5567
30	.2400	.3059	.3620	.4251	.4665	.5479

IL CASTELLO NOTTE SCURA

Il Castello notte scura è un gioco totalmente nuovo che si avvicina allo spirito delle «avventure» pur risultando sempre vario e coinvolgente ogni volta che viene giocato.

Lo scopo del gioco è triplo: inizialmente bisogna cercare di accrescere il numero e la forza dei soldati della propria armata. Poi bisogna trovare 5 chiavi che sono necessarie per entrare nel castello notte scura. Infine, una volta raccolte le chiavi, bisogna entrare nel castello e sconfiggere i suoi occupanti.

Le chiavi si possono trovare nei castelli o nelle tombe e si possono conquistare, insieme ad un cospicuo tesoro, dopo aver ucciso tutto i loro difensori.

In seguito ad ogni combattimento il luogo visitato sarà nuovamente riempito di mostri e di tesori.

I castelli e le tombe, insieme ad altri luoghi vari ed interessanti, sono sparsi in un territorio di 16x16 unità di lato.

Ogni unità è una possibile locazione che potete occupare. Sul video viene mostrata una sezione di 9x9 unità con la vostra armata sempre al centro. Ci sono 8 caratteri grafici speciali che sono:



= montagne



= deserto



= fiume



= città



= paese



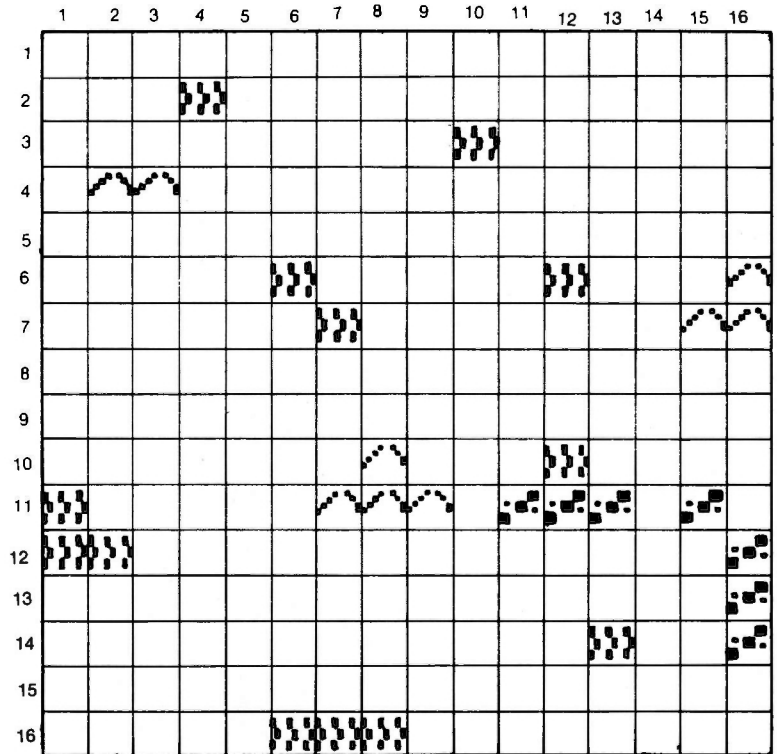
= castello



= castello N.S.



= tomba



Questa è la mappa che contiene le particolarità geografiche del campo di gioco e rimane invariata per tutte le partite. Si dovranno poi aggiungere le città, i castelli e le tombe che cambiano posizione ad ogni partita.

Nelle città e nei paesi (i cui mercati sono più cari di quelli cittadini), potete acquistare tutto ciò che serve allo svolgimento del gioco: armature ed armi, che accrescono la forza dei vostri uomini; informazioni sulle locazioni delle chiavi; cibo, che serve per nutrire l'armata ad ogni spostamento (l'acqua si trova solo nei fiumi); uomini, divisi in soldati e ladri (questi ultimi sono più cari ma permettono di derubare i castelli ed impossessarsi dei loro tesori senza affrontare i loro difensori). Quando acquistate cibo o uomini dovete indicare le quantità sempre con un numero di 2 cifre (es. per comprare 4 soldati dovrete premere 0/4).

Tabella riassuntiva degli acquisti possibili nei mercati

(i prezzi sono riferiti ai mercati cittadini)

- 1) CIBO: prezzo = 3 pezzi d'oro per unità per persona
- 2) INFORMAZIONI: prezzo = da 1500 a 3500 pezzi d'oro
- 3) UOMINI: soldati prezzo = 550 + equipaggiamento - ladri prezzo = 675 + equipaggiamento.

Armi e armature

	Prezzo	Fatica	Forza
Leggera	50	2	1
Media	200	5	3
Pesante	400	10	5
Daga	45	1	2
Ascia	200	2	5
Spada	350	5	7

Per calcolare la forza dei vostri soldati dovete sommare la forza dell'armatura + 1 (es. all'inizio con l'armatura leggera e la daga, la vostra forza

sarà $1+2+1=4$).

Il coeff. fatica influenza invece la fatica che farete spostandovi e si trova sommando la fatica delle armi e delle armature + 15.

Quadro riassuntivo delle caratteristiche dei nemici (tra parentesi la loro forza di combattimento)

	Aperto	Montagne	Deserti	Fiumi
Portatori tesori	Giganti (8)	Gnomi (5)	Predoni (11)	Tritoni (7)
Portatori pozioni	Vampiri (5)	Yeti (8)	Uomini sabbia (8)	Pirati (10)

NEI CASTELLI: SOLDATI (15)

NELLE TOMBE: SPETTRI (12)

NEL CASTELLO NOTTE SCURA: DRAGHI (forza variabile).

il 100%, può avere effetti letali sui vostri soldati. Pozioni e/o turni di riposo azzerano o diminuiscono l'affaticamento dell'armata.

Oltre che nei castelli e nelle tombe i nemici vi possono attaccare anche allo scoperto. Vi sono differenti tipi di nemici che si possono dividere in 2 classi: 1) portatori pozioni 2) portatori tesori ed eventualmente chiavi per il castello nottescura. Entrambe le classi possono portarvi in-

All'inizio del gioco viene però posto convenzionalmente a 15. Ad ogni combattimento e ad ogni spostamento aumenta la fatica che, se raggiunge

dubbi vantaggi ma bisogna attaccare solo se si è in superiorità numerica per non rischiare di essere sconfitti. La durata media di una partita (se riuscite a non farvi eliminare al primo combattimento), varia dalle 2 alle 4 ore e sono sicuro che anche voi, dopo i primi disastrosi tentativi, potrete entrare nel terribile "CASTELLO NOTTE SCURA".

LUCA MUZIO



LUNAR LANDER

Dopo tanti utility eccoci a voi con un gioco veramente notevole, sia per la grafica che per la dinamica.

È fuori dubbio che il gioco prende spunto dalla cartuccia Commodore dal titolo "Jupiter Lander", tuttavia ci sono alcune considerazioni da fare:

la prima è che Lunar Lander è eseguito tutto in Extended Basic, mentre l'originale è stato eseguito in linguaggio macchina, e che per ovvi motivi, di linguaggio appunto, il programma non offre la ZO-OMATA, cioè quando siete in prossimità di una coincidenza, l'originale ingrandiva il particolare in avvicinamento. Tuttavia possiamo dire che questo gioco offre veramente la possibilità di passare qualche ora di divertimento con il vostro TI 99.

IL GIOCO

Il gioco è molto semplice: la nostra astronave madre, di deposito sulla luna affinché con il vostro veicolo TERRARIA possiate trasportare i minatori lunari, che lavorano per estrarre il minerale di Galinoida della superficie del satellite, da una cava all'altra. Ovviamente questi minatori lavorano in luoghi impervi e scomodi da raggiungere. Poiché il tempo è denaro, la

ditta per cui lavorate vi dà un tempo limitato per il recupero, scaduto il quale, la vostra astronave, nel caso il recupero non fosse terminato, esploderà.

Ovviamente voi potete solo accelerare, a frenarvi penserà l'attrazione di gravità della luna, mentre l'atmosfera vi metterà in crisi per quanto riguarda gli spostamenti laterali.

È da notare che sulla superficie del pianeta, questi parametri sono soggetti continuamente a cambiamento per cui ci vorrà tutta la vostra abilità di pilota per portare a termine tutti i 16 quadri.

I COMANDI

Per giocare si hanno a disposizione 3 Moduli lunari. Il quadro non cambia mai tranne il colore delle montagne. Anche l'atmosfera e la gravità cambiano rispetto al quadro precedente.

Nel quadro a destra sono visualizzate le informazioni relative la gravità (GRAV), l'atmosfera (ATM), e il tempo (TM) e i moduli che restano ancora.

I minatori da recuperare sono quelle esili figure bianche che appaiono nel momento in cui il vostro modulo viene lasciato sulla superficie della luna. Per muovere il modulo

bisogna usare i tasti con le frecce tranne quello con la «X» che non serve. L'abilità sta nel guidare il modulo gas in modo da stare in aria senza esagerare e soprattutto senza toccare nulla, pena la perdita del modulo.

Nel caso usciate dal quadro verso l'alto, è da considerare irrimediabile la perdita del modulo. Quando sarete riusciti ad avvicinarvi ad un minatore sarà sufficiente la pressione del tasto «Q» per fare salire a bordo il minatore.

Saliti tutti i minatori, si passa ad un altro quadro.

Un piccolo consiglio prima dell'inizio: per non perdere tempo conviene salvare il minatore sulla cima all'estrema sinistra del modulo; dopo aver effettuato il recupero uscite sulla sinistra e ricomparite sulla destra, quindi recuperate il minatore in profondità e infine tutti gli altri. Buona fortuna!

MAURO MATTIAZZI

ROCKFELLER

In questo gioco voi impersonate un simpatico corvo che è stato imprigionato su una piramide di cubi colorati: scopo del gioco è quello di far saltellare il nostro eroe sulle facce superiori dei cubi fino a tingerle tutte di un solo colore.

Comunque Rockfeller non è il solo personaggio che saltella sui cubi: vi è Sam che ridiscende i cubi della piramide facendoli tornare al loro colore originario, per impedirglielo Rockfeller deve ingoiare Sam saltando sul suo cubo. Attenzione però, se è Sam a saltare sul vostro cubo sarete voi ad essere eliminati. Anche la palla verde, quando compare sullo schermo, va afferrata. Questo consente di bloccare il movimento di tutti gli altri personaggi presenti sulla piramide per un certo periodo di tempo. La palla rossa e quella nera vanno invece evitate, inoltre quella nera una volta giunta in fondo alla piramide fa uscire Coily, un serpentello avvolto su sé stesso che è l'unico in grado di risalire la piramide per dare la caccia a ROCKFELLER.

Quest'ultimo però ha un sistema efficacissimo per eliminare a sua volta Coily. Ai lati della piramide infatti si trovano due «dischi volanti»: sistemandosi su uno dei cubi

lateralmente della penultima fila, e spiccando un salto verso l'esterno, Rockfeller può saltare su uno di tali dischi e venire così trasportato fino in cima alla piramide, inoltre ciò consente di eliminare tutti i nemici presenti sullo schermo, Coily compreso. Dopo

gli schermi iniziali alle due piattaforme iniziali se ne aggiungono altre due, inoltre ogni piattaforma può essere utilizzata una volta sola per ogni fase di gioco:

Il movimento di Rockfeller avviene sempre in diagonale per cui il joystick viene utiliz-



zato in un modo diverso dal solito. Prendete il vostro joystick e mettetelo nella solita posizione, ora ruotatelo in senso orario di 45 gradi fino a formare un rombo con i suoi spigoli, in questo modo per far saltare Rockfeller in alto a sinistra basterà spingere il joystick in alto a sinistra e così per le altre direzioni. Se non avete il joystick o non volete usarli potete usare la tastiera utilizzando i seguenti tasti:

S — in alto a sinistra
D — in alto a destra
Z — in bassa a sinistra
X — in basso a destra

È molto importante assimilare al meglio i comandi per non commettere errori mortali: se Rockfeller salta per sbaglio dove non ci sono cubi, precipita nel vuoto e perde una vita.

SCHERMI

In ogni schermo occorre tramutare le facce superiori dei cubi nel colore finale; una volta completata una piramide, apparirà un nuovo schermo con una nuova serie di cubi.

Nei primi 3 schermi Rockfeller ha bisogno di un solo balzo su ogni cubo per passare dal colore iniziale (A) a quello finale (C). Negli schermi dal 4 al 6, il primo balzo di Rockfeller cambia il colore del cubo da uno stato iniziale A a uno intermedio B; per arrivare al colore finale C sarà poi sufficiente un secondo passaggio. Negli schermi dal 7 al 9, il colore passa direttamente da A a C dopo il primo passaggio per poi tornare ad A con il secondo. Per com-

pletare gli schermi in questa fase occorre quindi passare sui cubi un numero dispari di volte.

Negli schermi 10-12, il colore cambia da A a B a C, quindi nuovamente B, C, B, C e così via. Negli schermi 13-15, che si susseguono poi all'infinito, la faccenda si fa ancora più complicata: si raggiunge il colore finale C dopo due passaggi attraverso il colore iniziale A e quello intermedio B; basta però effettuare un balzo su un cubo con il colore C per tornare ad A per essere quindi costretti a ripetere i due passaggi precedenti.

Si ottengono 25 punti per ogni cubo che porta al suo colore conclusivo. Afferrare la palla verde dà diritto a 100 punti, mentre la cattura di Sam ne vale 300. Eliminare il serpente Coily dallo schermo con un disco fa guadagnare 500 punti, mentre il completamento di un intero schermo vale 3100 punti. I Rockfeller in dotazione sono 4: si perde una vita quando Rockfeller si scontra con uno dei suoi avversari oppure quando precipita nel vuoto. Si guadagna una vita-bonus ogni 5 schermi a partire dal 6. Alla fine del gioco per ricominciare bisogna premere RED o BACK. Un'ultima nota riguarda il fatto che è possibile sospendere momentaneamente il gioco semplicemente premendo il tasto P fino a quando compare la scritta PAUSA; per riprendere il gioco basta premere poi un tasto.

GIAMPAOLO BOTTIN

**ABBONATEVI
A TI 99
NEWSOFT.
CON SOLE
L. 95.000
RICEVERETE
11 NUMERI
DELLA RIVISTA
E IN PIU'
AVRETE
DIRITTO AGLI
SCONTI SUL
SOFTWARE
E
HARDWARE
OFFERTO DA
TI 99 MARKET**

Per abbonarsi
basta inviare
un assegno
non trasferibile
o un vaglia
postale a:

NEWSOFT SNC
Via S. Jacini, 4
20121 MILANO

ITALIANO

CONIUGAZIONE DEI VERBI REGOLARI IN FORMA ATTIVA CON GLI AUSILIARI SIA ESSERE CHE AVERE

Il programma vuole simulare il metodo di apprendimento e memorizzazione degli elementi necessari alla conoscenza ed alla padronanza dei verbi regolari nella forma attiva.

Esso si articola nella titolazione dalla linea n. 100 alla 150, quindi si passa alla introduzione ed alla lettura del verbo che si vuole coniugare, assegnato alla variabile H\$. Il verbo viene analizzato e scisso nelle due componenti: Radice e Desinenza, assegnate rispettivamente, a loro volta, alle variabili di stringa X\$ e Q\$.

A seconda della coniugazione selezionata si presenteranno dei menù differenti con colorazioni dello schermo diverse.

Tra le righe 260 e 270 si potrà inserire un goto 160, onde evitare che il computer accetti parole diverse da quelle terminanti in Are, Ere o Ire, maiuscole o minuscole che siano.

All'inizio di ciascuna sezione del programma, alle righe 270, 1630, 2660 alle variabili K\$ e J\$ vengono assegnate le desinenze singolare e plurale del participio passato. Segue la routine che visualizza lettera per lettera la scritta riguardante la coniugazione selezionata.

Per la prima coniugazione la scelta del tempo e del modo da coniugare avviene dietro la richiesta di scriverne esattamente la dicitura aggiungendo I per i tempi composti qualora il verbo richieda l'ausiliare ESSERE in quanto detto ausiliare concorre alla coniugazione della quasi totalità dei verbi intransitivi.

Per le altre due coniugazioni la scelta si opera su presentazione di un menù elencante tutti i tempi e modi e l'opzione per l'ausiliare Essere in seconda colonna. Il programma occupa tutta la memoria, per cui è impossibile aggiungere linee di istruzioni senza levarne altre.

Se si aggiungono le linee

151 GOTO 160

152 CALL CLEAR

153 PRINT "Non ho capito":

265 GOTO 152

si dovranno cancellare le linee di rem, nella speranza che l'esecuzione del programma non venga interrotta con il messaggio MEMORY FULL IN...

Questo procedimento renderà più elegante il controllo dell'INPUT iniziale.

Il fine didattico di diversificare il modo di accedere ai tempi e modi delle tre coniugazioni, è quello di allenare all'uso della tastiera ed alla

corretta ortografia oltre che alla corretta distinzione tra tempo e modo e tra verbi che richiedono l'ausiliare ESSERE invece che l'ausiliare AVERE.

Chi si meravigliasse nel vedere 6 righe di istruzioni nella linea dell'ON T GOTO, sappia che è possibile ottenere ciò scrivendo la linea di quattro righe nel modo più compatto possibile, battendo sull'ultimo spazio un carattere qualsiasi, premendo ENTER e quindi editando la linea appena scritta premendo SHIFT e Δ. Il cursore scivolerà fuori dall'ultima riga e quindi si potrà ancora scrivere sulla stessa linea.

Il programma tende a dimostrare come per saper coniugare con disinvoltura qualsiasi verbo, si debbano compiere due operazioni fondamentali: 1) saper riconoscere la coniugazione di appartenenza del verbo dalla desinenza, come fa il programma, 2) mandare a memoria tutte le desinenze.

Va da sé che volendo si possono visualizzare solo le desinenze, inserendo la sola desinenza ARE, ERE o IRE.

FERDINANDO GIGLIOTTI

CALORIE

Questo programma vi permette di sapere abbastanza rapidamente se i vostri pasti contengono troppe calorie, oppure se contengono in misura sufficiente tutte quelle sostanze che sono indispensabili per il buon funzionamento del nostro corpo, cioè quello che si chiama il bilanciamento della dieta.

Per prima cosa il computer vi domanderà che genere di attività svolgete; questo per stabilire il vostro fabbisogno giornaliero di calorie; ovviamente chi svolge un lavoro che richiede una grande attività fisica avrà un fabbisogno di calorie molto superiore di chi invece fa un lavoro sedentario.

Poi vi verrà chiesto se intendete prendere in esame solo un pasto oppure tutto il cibo ingerito durante il giorno.

Seguirà un elenco di alimenti, ad ogni domanda del computer occorre introdurre la quantità ingerita espressa in grammi ed è molto importante cercare di essere più precisi possibili sulla quantità.

Se un alimento non compare nel pasto è sufficiente premere ENTER per passare al seguente.

Terminato l'elenco degli alimenti comparirà un quadro riassuntivo con tutte le qualità totali delle varie sostanze chimiche presenti negli alimenti che sono stati introdotti nel programma.

Premendo ancora ENTER si

avranno le conclusioni sul bilanciamento, il computer cioè vi dirà quali sostanze risulteranno in eccesso e quali invece risulteranno in quantità inferiore alla minima giornaliera consentita.

Nel caso che alcuni minerali o vitamine risultino presenti in quantità troppo bassa, consultando la seguente tabella voi potrete sapere quali cibi dovrete introdurre nei vostri pasti per raggiungere un bilanciamento più adeguato.

I dati relativi alla composizione chimica degli alimenti sono stati tratti dalle tabelle del-

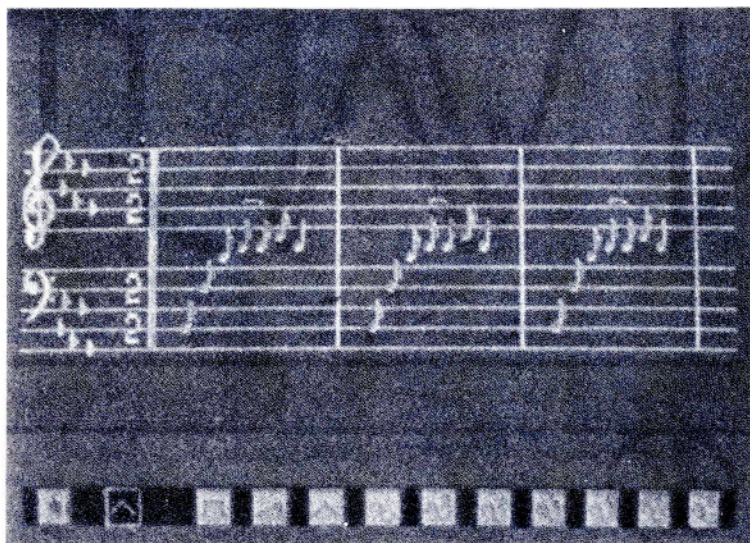


l'Istituto Nazionale per la Nutrizione.

MAURIZIO CAPPA

Elemento	Si trova in:
PROTIDI	carni, pesci, uova, latte, formaggi
GLICIDI	cereali, legumi, frutta, ortaggi
LIPIDI	burro, panna, lardo, olii
CALCIO	latte e formaggi
FERRO	frattaglie, tuorlo d'uovo, carne
POTASSIO	frutta, ortaggi, legumi, carni
VIT. B1	noci, castagne, legumi, fegato, rene, cuore, carne di maiale
VIT. B2	fegato, rene, cuore, latte, uova
VIT. PP	lievito, carne, fegato
VIT. A	fegato, rene, burro, formaggi, carote, pomodori, albicocche, pesche, mandarini

MUSICAL AIDER



Il programma in questione si può definire multiobiettivo in quanto diverse possono essere le finalità di impiego: diletto, utility, ecc.

Esso è principalmente diretto a tutti quei lettori, come il sottoscritto, che suonano preferibilmente ad orecchio e, pur disponendo di rudimenti di teoria musicale, faticherebbero non poco allorché si rendesse necessario individuare la esatta sequenza di note che costituisce una me-

lodia.

Tale operazione peraltro è fondamentale in tutti quei casi nei quali si desidera inserire colonne sonore in un programma autoprodotta che, va detto per inciso, acquista sempre un fascino particolare se è accompagnato da un adeguato commento musicale.

Nello stesso tempo va altresì offerta la possibilità di riascoltare la melodia digitata con vari timbri sonori, con di-

versa velocità, intervenendo con tutte le correzioni e modifiche del caso senza essere costretti a ripartire ogni volta dall'inizio.

IL FUNZIONAMENTO

Dopo il RUN appare il titolo accompagnato da una introduzione musicale (brano di una nota fuga) seguito da una breve descrizione sull'uso del programma.

La tastiera viene utilizzata alla stessa stregua della ottava di un pianoforte con i tasti egualmente disposti; si utilizzano infatti i tasti DO/SI corrispondenti alle lettere Z/M in basso a sinistra. I corrispondenti diesis sono collocati sulla linea di tastiera immediatamente superiore, come per i tasti neri di un pianoforte.

Generata la melodia (è possibile in qualunque momento modificare la ottava premendo i tasti 1/3 in quanto si dispone sino a tre ottave) si può scegliere fra diverse operazioni:

— Ripetizione del brano (tutto o in parte)

ANALISI LISTATO

N. Riga	Descrizione
130 - 310	Assegnazioni iniziali
320 - 460	Presentazione grafica
470 - 570	Presentazione sonora
580 - 690	Istruzioni per l'uso del programma
700 - 840	Introduzione ottava desiderata
850 - 1330	Attivazione tastiera
1340 - 2160	ROUTINE comprendente la selezione funzione e l'ascolto
2230 - 2400	ROUTINE di stampa (o pagina video)
2410 - 2510	ROUTINE CAMBIO PAGINA
2540 - 2860	ROUTINE transcodifica tasti/frequenze.

VARIABILI PRINCIPALI

GR\$(13)	Identificazione grafica nota
OT\$(3)	Identificazione grafica ottava
M(250)	Frequenza nota memorizzata
D(250)	Durata nota memorizzata
O	Ottava prescelta
U	Numero battute
V	Timbro prescelto in ascolto
NOT(13)	Codifica tasto digitato
Z(50)	Codifica interna

— Modifica del brano
— Stampa della sequenza del tipo di nota, della frequenza, della durata.

In fase di digitazione della melodia, va aggiunto che è possibile in qualunque momento correggere la ultima o le ultime note introdotte premendo il tasto 9 e ripetendo la battuta; la durata di pressione dei tasti è memorizzata. La audizione è possibile su 5 diverse velocità e con diversi timbri:

- monotonale
- fisarmonica
- organo classico
- organo moderno

È così possibile riascoltare, anche se si è suonatori impacciati che fanno molti errori, la melodia introdotta, con perfetto tempo e sequenza.

La stampa della sequenza delle note, con relativa frequenza e durata, è particolarmente utile per introdurre, in una istruzione di DATA il commento musicale a programmi da noi stessi prodotti.

Per quanto riguarda le caratteristiche del programma, si rimanda alla allegata tabella, sottolineando solo che il largo impiego di vettori (necessario per velocizzare i vari controlli), lascia disponibile un Kb. circa. Una apparente complicazione del programma, con le sue transcodifiche interne, è una conseguenza di ciò.

R. DRUSIANI

*Avete sviluppato
un programma
originale,
divertente, utile?
Volete vederlo
pubblicato
su uno dei
prossimi numeri
di*

**TI-99
NEWSOFT?**

*Leggete allora
con attenzione
a pag. 31-32
e mandateci
le vostre
proposte*

VENDO - COMPRO - CAMBIO - SCAMBIO

VENDO

Vendo TI 99/4A + coppia joystick + due fascicoli con programmi + 2 cassette «SOCCER» e «PARSEC» + cavetto registratore + cassetta con 8 lezioni a sole L. 300.000. Tel. 02/304234 (ore pasti).

TI 99/4A accessoriato di Joystick filo registratore moduli 1) Alpiner 2) Tical 3) Indo-ov. soccer 4) Othello 5) Blackjack and Poker. Schede funzionanti di numermagic, adventure, challenger. EX. BASIC (non perfetto + 80 programmi funzionanti registrati su cassetta). Andrea Sebastiani - Via Lucandi 11 - 02100 Rieti - Tel. 0746/43380. (ore 14,00-14,30, 21,00-21,30) giorni feriali.

TI 99/4A vendo completo di cavi, Trasformatore, Modulatore, Manuale in italiano, SSS Tombstone CITY, numerose cassette registrate e abbonamento annuale a TI99 NEWSOFT. Tutto nuovo. L. 350.000. Testoni Paolo - 44040 Dodici Morelli (FE) - Tel. 051-971314 al pomeriggio.

Vendo: traduzione in italiano del manuale della Mini Memory e del Line-by-Line assembler. L'interessato può allegare alla richiesta un vaglia di lire 30.000. Leucci Claudio - Via Putignano - S. Ermete, 288/1 56100 Pisa.

Vendo **TI Terminal Emulator II** ancora imballato a L. 50.000 e scambio programmi per TI 99. Grata Giuseppe - Via Fornaci, 1 - 25030 Torbole C. (BS).

Vendo **TI-NY LOGO** su cassetta per TI99/4A in italiano (funziona con la sola console) al prezzo eccezionale di L. 35.000 + spese. Cassetta con Bioritmi, Filetto, ecc. L. 20.000; Cassetta simulazione di mercato L. 15.000. In blocco L. 60.000 + spese. Friggi Luciano - Via M. Magnini 103 - 06100 Perugia - Tel. (075) 753551 (h. 20).

Vendo **TI 99/4A** ottime condizioni due mesi di vita + cavetti registratore + manuale d'uso + A scuola con il TI 99/4A + 10 giochi su cassetta L. 260.000. Filippo Conti - Via Milliaro 6 - Bologna 051/382469 qualsiasi orario.

Vendo computer **TI 99/4A** accessoriato di tutto + coppia di joystick + seguenti moduli SSS: Blasto (gioco), Mini Memory (con assembler lyne by lyne), Extended Basic + cassette con vari programmi. Il tutto a L. 500.000. Simoncini Lorenzo - Via Torelli n° 55 - 43100 Parma - Tel. 0521/45693.

Vendo **modulo SSS:** Hunthe Wumpus, due cassette (della serie Adventure) Adventureland e Ghost Town, programmi e listati (anche in Extended) più di 150. Oppure cambio il tutto con moduli SSS: Alpiner, Mini Memory, speech editor (con eventuale aggiunta di denaro) tratto con Torino-Piemonte. Luca Carazza - Via Lodi 2 - 10152 Torino - Tel. 237041.

Vendo: aiuto alla programmazione n° 1 (ZND Ascii sel, caratteri minuscoli, definizione caratteri, 1/0 subroutines); imparare da soli il Basic Esteso, corso di TI Basic; oldies but codiesgame. Tutti su cassette TI originali. Vendo o Cambio con joystick o moduli SSS. Badiali Giampaolo - Via C. Pisacane 12 - 60033 Chiaravalle (AN) - ore serali (dopo le 20) - Tel. (071) 946311.

Compro per **TI 99/4A** modulo Extended Basic completo di manuale e coppia di joystick. Aioldi Silvia - Via Agucchi n° 8 - 40131 Bologna - Tel. (051) 387744

Compro manuale Editor Assembler a prezzo ragionevole (sarei anche disposto a fare le fotocopie). Scrivere o Telefonare dopo le 14 a Maionchi Federico - C.so Racconigi 151 - 10141 Torino - Tel. 011/389512.

Compro registratore dati Texas se in buono stato e a prezzo modico. Rossi Giovanni - Via P. Ricasoli n° 63 - 50127 Firenze - Tel. 4377271.

COMPRO

Cerco modulo Extended Basic per TI 99 offro L. 100.000. Botto Livio - Via Ressa n° 23 - Mondovì (CN) - Tel. 0174/44403 ore pasti.

Compro modulo Extended Basic per TI 99/4A. Contatto solo possessori in Roma. Scrivere o telefonare a: Taraddei Sandro - Via Trionfale n° 8204 - Tel. 3370210 - dopo le 20 - Roma.

Cerco Personal Record Deeping + cavetto per registratore + coppia joystick, a prezzi ragionevoli. Paolo Riffellato - Via G. Pascoli 9 - 35020 Ponte San Nicolò (PA) - Tel. 049/717325 (20-21).

Cerco Extended + cassetta «imparate l'Extended + espansore di memoria 32 K. Prezzo ragionevole. Compro anche separati. Tratto solo con Milano e prov. Mantene Massimo - Via Appennini 67 - 20151 Milano - Tel. 3532367 (ore 20,00).

Compro **modulo SSS Extended Basic** completo di documentazione (opzionale) interessato a coppia Joystick prezzo modico. Luciano Emiliano - Via A. Ginma 243 - Tel. 080/231989 - Bari ore pomeridiane.

Cerco urgentemente per TI 99/4A disk Drive Card + disk control card + Mini Memory + terminal Emulator editor Assembler a buon prezzo. Rizzo Giuseppe - Via Radio Asa 210 - Mondragone (CE) - Tel. 0823978004 pre 19,30-21,30.

CAMBIO

Cambio chitarra elettrica + amplificatore + pedale ritmico con modulo SSS Editor Assembler completo. Sergio Vittone - Via Moriondo n° 46 - Buttiglieria d'asti (AT) - Tel. 9871556 (prefisso 011)

Cerco possessori di Minimemory interessati al linguaggio Assembler e L/M, scopo scambio informazioni e Software. Possiedo fantastici programmi su cassetta per minimemory, un assembler che permette di utilizzare tutti i 4 Kbyte della M.M. Viviani Gabriele - Via Garesio 24/6 - Cap. 10126 Torino - Tel 011/676290.

SCAMBIO

Scambio programmi in Basic ed in Extended originali e non con Torino-Milano-Genova. Telefonare o scrivere a Fantino Franco - Via Rieti 10 - 10095 Grugliasco (TO) Tel. 011/704313 orario: dalle 19 alle 22.

VENDO

Vendo espansione memoria 32 K collegabile direttamente al TI-99, modulo mini-memory, modulo Assembler con manuali originali in inglese. Procuo direttamente dagli USA altri Sort. Tel. ore pasti serali al 045/528738. Daniele Consonni - Via Alberti - 37132 Verona.

Cambio programmi su cassetta con altri programmi su cassetta. Scrivere a: Di Pisa Filippo, Via Riva Di Reno 4 - Bologna - Tel. 051/554520 ore 3-4 - P.S. vendo programma piano a L. 50.000.

Qualsiasi prova vendo TI 99/4A completo + Registratore Data Recorder + SSS TI Extended Basic + SSS Video chess + SSS Personal Record Kiping + 2 manuali + Joystick a L. 500.000. Regalo software su cassetta. Giaccaglia Marcello - C. Belgio 104 - Torino - Tel. 011/898428 ore pasti.

Vendo TI 99/4A + Cavetto integrale + Registratore + Manuali + 3 casseyye giochi vari + Modulo giochi scacchi. Tutto a L. 400.000 trattabili. Telefonare a Lino 3556391 ore serali.

Vendo TI 99/4A, completo di: Consolle + Alimentatore + Modulatore TV + Cavo per 2 registratori + Circa 40 programmi su cassetta + TV b/n 26 pollici a L. 350.000. Occhielli Roberto - Via Uruguay 32/F - Tel. 3081866 - Sabato mattino possibilmente.

Vendo 5 moduli gioco (Parsec, Moon Mine, The Attack, Cars Wars, Chislom Trails), il tutto al prezzo di L. 150.000 trattabili, vendo i moduli anche singolarmente. Telefonare o scrivere a Stefano. Pinni Stefano - Via Bernardino Verro 78/F - Tel. 02/8462658 - 20141 Milano ore 20.30-21.30.

Programmi per TI 99/4A in Basic ed Extended riguardanti Utilities e giochi. Vendo inoltre programmi musicali di mia programmazione. Per informazioni: Antonio Gianni - Via Poggio 9 - Vazia (Rieti) - Tel. 0746/72322.

Vendo coppia Joystick originali Texas a L. 35.000 e schemi elettrici di consolle, alimentatori, box, espansione 32K, Speech Synthesizer, interfaccia RS-232 a L. 25.000 (spese di spedizione incluse). Scarpati Cioffari Carlo - Via Garessio 24/10 - Tel. (011) 6963574 - 10126 Torino.

Vendo TI 99/4A, Espansion Box, Ram Expansion Card Disk Drive Card, Disk Drive, Speech Synthesizer, Extended Basic, Editor Assembler, con schemi elettrici originali Texas, Data Books e circa 300 programmi su disco. Pelva Zoltan - Viale Toscana 24 - 36061 Bassano (VI) - Tel. 0424/26872 sabato e domenica.

Vendo TI 99/4A condizioni perfette per passaggio sistema superiore + Alpiner, Indoor Soccer, Othello, TI-Calc, Black Jack and Poxer + Joystick (coppia) (vendo anche separatamente) Andrea Sebastian - Via Ivcardi 4 - Tel. 0746/43380 (Rieti) ore 14.00-21.00.

2 Programmi: gestione Totocalcio (Ex. B). Per 1 - Picchetto e ottimizzazione del sistema, quindi si evita di giocare DP e TR inutili. Per 2 - Sviluppo di sistemi integrali, integrali derivati, statistici + sezioni. Non ha limiti x le colonne da elaborare. I 2 programmi L. 40.000 - 1 L. 23.000 contrassegno compreso.

2 programmi gestione Totocalcio (Ex. B.) 1 - Picchetto e ottimizzazione del sistema, quindi senza sprechi di DP e TR. 2 - Sviluppo di sistemi integrali, derivati, sezioni - Non ha limite x le colonne da elaborare - I 2 programmi L. 40.000 - 1 L. 23000 contrassegno compreso. Enrico Mollo - Via Martiri, 23/A - 12014 Demonte (CN) - Tel. 0171/95258.

Vendo TI 99/4A + Collegamenti + Manuale + Libro per TI-99 dal titolo: "Dinamic Games for your TI 99/4A" + Cartidge: Soccer e Ti Invaders + Corso di Basic su cassetta + Tantissimi programmi (giochi Utility, Matematica), il tutto perfetto, come nuovo ad un prezzo impensabile. L. 300.000 tutto. Per informazione rivolgersi a: De Mare Stefano - Via B. Disi 29 - 00165 Roma - Tel. 6225401 - Ripondo a tutti.

downloaded by www.ti99iuc.it

CAMBIO

Cambio programmi su cassetta o semplicemente dei listati. Chiedere di Carlo. Mecconi Carlo - Via Collatina 76 - Tel. 2594394 - Orari tra le 17.30-18.30.

Scambio giochi su cassetta in TI Basic e Extended con altri equivalenti dispongo di giochi con buona grafica assicuro risposte a tutti!!! Matteini Alessandro - Via Albano Sorbelli, 10 - Roma - Tel. 3390891 (06) Cap. 00135.

Regalo Modulo SSS Music Maker a chi mi vende a prezzo modico Modulo SSS Editor Assembler + Dischetti + Manuale ecc. Possibilmente zona Torino e dintorni. Sergio Vittone - Via Moriondo 46 - Buttigliera d'Asti - Tel. 011/9871556 ore pasti.

A Roma è nato il 99ers Club. Disponiamo di Software in qualsiasi linguaggio. Per informazioni o modalità d'iscrizione telefonare. Fabio / Federico - Tel. 06/789392-9456079 esclusi festivi.

Crechiamo utilizzatori di TI 99/4A + Mini-memory scopo scambio idee e informazioni. Vendiamo o scambiamo con Software equivalente magnifici programmi in L/M e un assemblatore in Basic che permette di utilizzare interamente i 4K Byte della Minimemory. Telefonate o scrivete a: Viviani Gabriele - Via Garessio 24/6 - 10126 Torino - Tel. 011/676290.

MODULO PICCOLI ANNUNCI

TI 99 MARKET

☐ VENDO☐ COMPRO☐ CAMBIO☐ SCAMBIO

Desidero pubblicare il seguente annuncio:

.....

.....

NOME - INDIRIZZO - TELEFONO - ORARI:

.....

La pubblicazione è riservata solo a chi utilizza il presente tagliando ed è gratuita.

SPEDITE IL TAGLIANDO A: TI 99 NEWSOFT - EDISOFT - Via S. Jacini, 4 - 20121 MILANO

VIZI DI GIOVENTU'

Spettabile redazione, sono un vostro affezionato lettore e vi seguo fin dal primo numero uno (o zero!!). Ed è a proposito di ciò che vi scrivo questa mia per avere, almeno stavolta, alcuni chiarimenti.

Sul numero due della rivista (uno di copertina, due di cassetta) un vostro inserto pubblicitario avvisava della già avvenuta uscita di un numero zero. Inoltre era inserito un modulo di richiesta listati e un questionario che, se rispedito debitamente compilato, sarebbe stato ricompensato con degli "adesivi a colori di TI-99 NEWSOFT". Già entusiasta della vostra rivista, mi affrettai a scrivere una lettera per richiedervi il numero zero e inserirvi il modulo di richieste listati ed il questionario. Dopo aver atteso fiducioso circa due mesi, finalmente mi arriva un pacco da parte di TI-99 NEWSOFT. E qui cominciano le sorprese.

Dopo averlo aperto ho notato l'assenza sia degli adesivi che dei listati.

Poco male, penso, mi arriveranno in seguito. Ma perché poi non li hanno uniti alla cassetta? Ma la più grossa delusione è venuta quando ho visto che la cassetta nient'altro era che il numero uno da me già acquistato presso il mio edicolante.

Dunque la numerazione della cassetta non era coerente con quella riportata sulla rivista. E questo errore è stato ripetuto fino alla uscita del numero 3B, che suppongo sia servito per correggere tale

incoerenza. Non vi pare dunque che quella pubblicità fosse un tantino ingannevole? E che ne è dei listati e degli adesivi da voi promessi e a tutt'oggi non ancora arrivati? Peraltro tutto ciò mi ha indotto a tenermi, se pur ancora imballata, la cassetta, di cui non so che farmene, dato che avendo già pagato 12.900 lire, una eventuale rispeditizione al mittente avrebbe comportato più costi che eventuali ricavi.

Complimenti!! Però che ne direste di mettere un po' più attenzione nella scrittura di programmi della rivista e nella registrazione del nastro? Infatti spesso molti di essi vanno in errore, o fermandosi, o generando elaborazioni non volute. Vi cito uno dei tanti esempi: Othello. Se a partita iniziata viene premuto il tasto I (inizio) effettivamente riparte dall'inizio ma il gioco impazzisce. Tale errore è stato rilevato giocando contro il computer. Alcuni programmi ancora presentano svisse così evidenti quasi non fossero mai stati testati. È il caso di "agenda telefonica": osservate attentamente il menù...

Gli errori tipografici poi... O quelli dovuti al nastro difettoso o registrato inversamente rispetto alla facciata? (N 3B). Chi vi scrive è un programmatore di professione come voi e quando torna a casa non vorrebbe continuare a farlo per mettere a posto programmi che non funzionano bene. Spero siano vizi di gioventù, in quanto la vostra rivista rimane un validissimo punto fermo per tutti coloro che vogliono continuare ad

utilizzare il TI-99 con i programmi da voi forniti (un po' troppi giochi!).

Fiducioso in una vostra risposta alla mia lettera colgo l'occasione per porgervi distinti saluti e auguri per un sempre crescente successo della vostra rivista.

GIUSEPPE GIACO'
Bari

Pubblichiamo volentieri la lettera del signor Giaco perché pensiamo rispecchi i sentimenti e l'opinione di molti nostri lettori.

Certo, caro signor Giaco, Lei ha ragione quando dice che alcune cassette risultano difettose; in questo caso, per Lei e per tutti coloro a cui dovesse succedere, basta rispedirle a noi, spiegando brevemente dove sta l'errore, e noi le sostituiamo.

D'altra parte è inevitabile che su migliaia di cassette prodotte qualcuna risulti imperfetta.

Come lei giustamente sottolinea, siamo una rivista giovane, ancora in fase di "rodaggio", e quindi vogliate perdonarci le imprecisioni e i ritardi: l'importante è che un numero sempre crescente di abbonati e lettori ci stia dimostrando come la nostra iniziativa sia valida e venga apprezzata, così come i nostri sforzi per migliorare o offrire ogni numero qualcosa di più a tutti i "novantanoviani".

Per i listati e gli adesivi a colori di TI-99 NEWSOFT preghiamo tutti coloro che ce ne hanno fatto richiesta di pazientare ancora un po'.

COLLABORATE CON NOI

Se avete scritto un programma che vi sembra adatto per essere pubblicato su TI-99 Newsoft, speditecelo allegando questo formulario compilato in ogni sua parte per consentirci di valutarlo e di darvi subito la nostra opinione. Preferiremmo ricevere sia il nastro che il listato, ma possiamo lavorare anche solo sul nastro. Per ovvi motivi di organizzazione, vi avvisiamo che non restituiamo il materiale pervenuto in redazione. Ci serve inoltre una breve descrizione del programma di almeno due cartelle dattiloscritte, una breve analisi del listato ed ogni altra documentazione che riterrete utile (disegni, flow chart, illustrazioni, etc.).

Accertatevi che su ogni cosa che ci spedite, anche sulla cassetta, sia riportato il vostro nome e tipo di programma. Tutto questo ci sarà di aiuto per classificare il vostro programma durante le nostre prove e ci permetterà inoltre di avere tutte le informazioni necessarie per la rivista. downloaded by www.ti99iuc.it

Se non volete tagliare la rivista potete fare una fotocopia di questa pagina. Questo formulario apparirà, comunque, regolarmente da ora in poi in ogni numero di TI-99 Newsoft.

Ogni programma pubblicato sarà retribuito secondo la qualità a partire da L. 60.000.

DATI PERSONALI

NOME
COGNOME
VIA N°
CITTA' C.A.P. PROV.
TEL.

SCHEDA SOFTWARE

TITOLO DEL PROGRAMMA

SE GIRA SU ALTRI COMPUTER INDICARE QUALI

KBYTES DI MEMORIA RICHIESTI

PERIFERICHE NECESSARIE (joysticks, Extended Basic, etc....)

.....

TIPO DI PROGRAMMA (specificare se originale)

ISTRUZIONI DI CARICAMENTO

BREVE DESCRIZIONE DEL PROGRAMMA

.....

.....

.....

RISERVATO ALLA REDAZIONE

DATA DI RICEVIMENTO

☐ DA RISPEDIRE ALL'AUTORE PER MIGLIORAMENTI

DATA DI SPEDIZIONE

☐ PUO' ESSERE PUBBLICATO NEL N. DI DI TI-99 NEWSOFT

VALUTAZIONE

☐ DA PUBBLICARE

☐ RICHIEDE ALCUNE MODIFICHE

☐ NON PUBBLICABILE

☐ GIA' PUBBLICATO DA ALTRE RIVISTE

☐ NON GIRA

☐ NON SI RIESCE A CARICARE

TI 99 MARKET

Siamo finalmente in grado di offrirvi i seguenti materiali. Tutti i prezzi sono comprensivi di IVA ed escludono i costi di spedizione e imballo contrassegno.

DESCRIZIONE	PREZZO	PER GLI ABBONATI
HARDWARE		
Home Computer TI 99/4A - 16bit - 16 KB completo di alimentatore e modulatore per collegamento TV con 2 moduli SSS gioco/educativi	229.000	200.000
Cavetto di collegamento al registratore	12.000	10.000
Cavetto adattatore Joysticks di qualsiasi tipo (Spectravideo...)	29.000	25.000
Joystick singolo	22.000	19.000
Interfaccia parallela esterna per il collegamento di stampanti	200.000	180.000
Cavo di collegamento interfaccia parallela esterna con qualsiasi stampante tipo Centronics mt. 1	36.000	33.000
Stampante Seikosha GP500A matrice 5x7, 50 crt/sec 80 colonne, normale ed espanso, grafica	549.000	499.000
Stampante EPSON RX80, 100 crt/sec grafica, 80/132 colonne bid. ott.	759.000	699.000
Stampante EPSON RX80 F/T come sopra gestisce anche foglio singolo	869.000	790.000
Monitor colori Hantarex CT900/2 con audio, 15" e cavo collegamento	589.000	530.000
Espansione di memoria esterna (non richiede il BOX) di 32 Kb ram downloaded by www.ti99iuc.it	200.000	180.000
SOFTWARE		
TI LOGO in Italiano - Mondadori	179.000	162.000
Modulo EXTENDED BASIC	180.000	160.000
Modulo TERMINAL EMULATOR	49.000	42.000
Modulo PRG gest. dati personali	49.000	42.000
Modulo STATISTIC (tedesco)	19.000	16.000
Modulo ADD-SUB A/1 - educativo	19.000	16.000
Modulo MATH GAMES 6 - educativo	19.000	16.000
Modulo NUMERATION 1 - educativo	19.000	16.000
Modulo DEMOL. DIVISION - educativo	19.000	16.000
Modulo MATH GAMES II - educativo	19.000	16.000

segue

DESCRIZIONE	PREZZO	PER GLI ABBONATI
segue SOFTWARE		
Modulo EARLY READING - educativo	19.000	16.000
Modulo WUMPUS - gioco	19.000	16.000
Cassetta VIRGIN ROBOPODS	16.000	13.000
Cassetta VIRGIN FUN PAK	16.000	13.000
Cassetta VIRGIN FUN PAK 2	16.000	13.000
Cassetta VIRGIN FUN PAK 3	16.000	13.000
Cassetta BLUEGRASS corsa ippica	10.000	8.500
Cassetta DECATHLON	12.000	10.000
Cassetta GOBLINS REVENGE labirinto	10.000	9.000
Cassetta SNOW TREK corsa di slitte	10.000	9.000
Cassetta UP PERISCOPE batt navale	10.000	9.000
Cassetta FAMILY GAME 4 giochi	10.000	9.000
Cassetta OLDIES BUT GOODIES GAME	10.000	9.000
Cassetta PERSONAL FINANCIAL AIDS	10.000	9.000
Cassetta TI 99 CALC	30.000	27.000
Cassetta TOMBOLA + GIROTONDO	30.000	27.000
Cassetta ELEUSIS + 1000 NOMI	30.000	27.000
DISCO RADAR	30.000	27.000
DISCO 2000 COW BOY	30.000	27.000
DISCO SNOOPY	30.000	27.000
DISCO QUADRO X	30.000	27.000
DISCO SCOPA	30.000	27.000
DISCO P.A.G.A.	30.000	27.000
OFFERTE KIT		
TI LOGO Mondadori in Italiano e espansione di memoria esterna da 32 Kb con libro:		
Impariamo a programmare in TI Basic	369.000	330.000
4 moduli da lit. 19.000 assortiti	59.000	54.000
9 cassette assortite	99.000	90.000
Stampante SEIKOSHA GP500A completa di cavo ed interfaccia parallela	699.000	660.000